VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ
DEPARTMENT OF INFORMATION SYSTEMS

ŠTATISTICKÁ ANALÝZA DÁT Z PDF SÚBOROV

STATISTICAL ANALYSIS OF DATA FROM PDF FILES

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **KRISTÍNA OLTMANOVÁ**
AUTHOR

VEDOUCÍ PRÁCE **Ing. VLADIMÍR BARTÍK, Ph.D.**
SUPERVISOR

BRNO 2021

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav informačních systémů (UIFS) Akademický rok 2020/2021

Zadání bakalářské práce

Studentka: **Oltmanová Kristína**
Program: Informační technologie
Název: **Statistická analýza dat z PDF souborů**
Statistical Analysis of Data from PDF Files
Kategorie: Data mining

Zadání:

1. Seznamte se s problematikou statistické analýzy dat a získáváním znalostí z dat.
2. Seznamte se s programovacím jazykem Python, prostudujte dostupné knihovny pro zpracování PDF souborů a statistickou analýzu (regulační diagramy, data mining, ...)
3. Po konzultaci s vedoucím navrhněte demonstrační aplikaci provádějící vybrané metody statistické analýzy s daty extrahovanými z PDF souborů.
4. Navrženou aplikaci implementujte a proveďte testování na vhodném vzorku dat.
5. Zhodnoťte dosažené výsledky a další možnosti pokračování tohoto projektu.

Literatura:

- Jarošová, E., Noskievičová, D.: Pokročilejší metody statistické regulace procesu. Grada, 2015. ISBN 978-80-247-5884-8.
- Layton, R.: Learning Data Mining with Python. Packt Publishing, 2015.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1-3.

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce: **Bartík Vladimír, Ing., Ph.D.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2020
Datum odevzdání: 12. května 2021
Datum schválení: 22. října 2020

Abstrakt

Táto práca sa zaoberá problematikou získavania dát z tabuliek dokumentov vo formáte PDF a ich následnou analýzou s využitím štatistických nástrojov. Cieľom práce je demonštrovať proces získania, spracovania a vyhodnocovania dát na dopredu stanovenej vzorke dokumentov typu PDF, ktoré z hľadiska programového spracovania tvoria konečnú množinu podskupín so spoločnými vlastnosťami. Práca najskôr predstavuje základy spracovania PDF súborov a základné matematické princípy, ktoré sú potrebné k zhodnoteniu štatistických parametrov získaných dát. Získané teoretické princípy sú následne uvedené do praxe a do programovej podoby v programovacom jazyku Python. Výsledná webová aplikácia je naprogramovaná s využitím knižnice Flask a je použiteľná na lokálnom serveri.

Abstract

This thesis is concerning the process of data extraction from tables from documents in PDF format and their subsequent analysis with the exploitation of statistical methods. The goal of this thesis is to demonstrate the process of obtaining, processing and analyzing data from PDF files, which, in consideration of their program processing, create a finite number of subgroups with common characteristics. Firstly, the reader will become acquainted with the fundamentals of PDF file processing and basic mathematical principles that are required in order to statistically evaluate given data. Obtained theoretical principles are then applied to practical use and programming form in the Python programming language. The resulting web application is programmed using the Flask Python library and is usable on a local server.

Kľúčové slová

regulačný diagram, štatistická regulácia procesu, Shewhartov regulačný diagram, Hotellingov regulačný diagram, index spôsobilosti procesu, extrakcia tabuliek z PDF, štatistická analýza, Python, Flask, webová aplikácia

Keywords

control chart, statistical process control, Shewhart control chart, Hotelling control chart, process capability index, PDF table extraction, statistical analysis, Python, Flask, web application

Citácia

OLTMANOVÁ, Kristína. *Štatistická analýza dát z PDF súborov*. Brno, 2021. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Vladimír Bartík, Ph.D.

Štatistická analýza dát z PDF súborov

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracovala samostatne pod vedením pána Ing. Vladimíra Bartíka, Ph.D.. Uviedla som všetky literárne pramene, publikácie a ďalšie zdroje, z ktorých som čerpala.

.....
Kristína Oltmanová
10. mája 2021

Poďakovanie

Na tomto mieste by som chcela poďakovať vedúcemu bakalárskej práce, Ing. Vladimírovi Bartíkovi, Ph.D., za usmernenia a cenné rady, ktoré mi pomohli k pochopeniu danej problematiky a spísaniu tejto práce.

Obsah

Úvod

V súčasnej dobe sú PDF súbory samozrejmosťou takmer pre každého. Človek totiž nemusí byť počítačovým expertom, aby sa so súborom tohto typu stretol v každodennom živote. K všeobecným zručnostiam priemerných používateľov technológií neodmysliteľne patrí čítanie, prezeranie a ukladanie dokumentov z iných programov práve do tohto formátu.

Úprimne povedané, ja som práci s nimi nikdy neprikladala obzvlášť významnú úlohu. O to väčšie bolo moje prekvapenie, keď som sa dozvedela o existencii mnohých služieb, ktoré ponúkajú export dát alebo konverziu PDF do iných formátov. Tieto služby mali dokonca samostatnú sekciu na väčšine známych *freelance* stránok, z čoho vyplýva, že dopyt po týchto službách musí byť naozaj vysoký. Osobne by som nikdy nepovažovala túto oblasť za tak veľmi žiadanú. Toto vo mne vzbudilo prvotnú vlnu záujmu, ktorá napokon viedla až k výberu témy mojej bakalárskej práce.

V tejto bakalárskej práci sa budem venovať štatistickej analýze dát získaných práve z PDF súborov, konkrétne z tabuliek. V úvodných kapitolách je bližšie predstavená problematika extrakcie dát, ktorá vysvetľuje prečo získavanie dát z tohto typu súborov nie je až také jednoduché ako sa na prvý pohľad môže zdať.

Nasleduje časť zameraná na štatistickú reguláciu procesov, ktorá obsahuje potrebné matematické a štatistické predpoklady, na základe ktorých budú dáta štatisticky analyzované.

V poslednej časti je podrobne popísaný proces návrhu a implementácie výslednej webovej aplikácie. Sú tu bližšie špecifikované očakávané typy súborov, návrh grafického prostredia ako aj využité technológie.

Cieľom práce je spracovať dodanú množinu súborov typu PDF, teda získať z nich potrebné numerické dáta a uložiť ich do takej podoby, ktorá uľahčí ďalšiu prácu s nimi. Dáta, ktoré už sú uložené vo vhodnej podobe je následne možné štatisticky analyzovať pomocou vykreslenia regulačných diagramov, ktoré vypovedajú o prípadných podozrivých hodnotách, ktoré sa štatisticky vymykajú normálu.

Výsledná aplikácia slúži na zjednodušenie a uľahčenie spracovania tabuliek z dodaných typov PDF dokumentov.

Kapitola 1

Proces získavania dát z PDF súborov

Táto kapitola poskytuje úvod do problematiky extrakcie dát z PDF súborov a ponúka prehľad dostupných riešení, ktoré sa v dnešnej dobe na extrakciu dát používajú. Bližšie predstavuje samotný formát PDF a jeho súradnicový systém. Následne ponúka predstavenie základných princípov získavania dát z PDF súborov ako aj prehľad používaných technológií v prostredí Python.

1.1 Motivácia získania dát z PDF súborov

Dokumenty typu PDF sú celosvetovo rozšírené a uznávané ako vhodný formát na šírenie a distribúciu obsahu. Tvoria aj tradičný formát pre zaznamenávanie meraní z prístrojov v oblasti výroby, pričom do tohto typu dokumentu bývajú dáta uložené a následne zdieľané za účelom ďalšieho manuálneho vyhodnotenia. Tento zaužívaný postup ráta s ľudským faktorom, avšak s pokrokom a neustálymi inováciami sa ponúka potreba hľadania riešenia, ktoré skĺby staré postupy prezerania dát vo formáte PDF a zároveň zaistí export týchto dát do formy, ktorá môže byť ďalej strojovo spracovaná.

PDF súbory boli už spomínané ako exportný výsledný formát meraní. Ich výhoda spočíva v tom, že viacero zabehnutých spoločností v oblasti priemyslu a výroby, ktoré tieto merania vykonávajú na pravidelnej báze, už majú svoje systémy prispôsobené na zaobstaranie dát v rôznych formátoch a ich následný prevod do PDF. Možno práve takto uchovávajú aj archívne merania, ktoré majú záujem podrobiť analýze.

Všetky spomínané účely by mohli byť urýchlené a zjednodušené automatizáciou prevodu potrebných dát z ukladaných PDF súborov.

Je potrebné si uvedomiť, že jednoduchý text v odstavcoch v problematike extrakcie dát zo súborov typu PDF nepredstavuje problém. Je všeobecne známe, že z klasického digitálneho súboru PDF je možné text skopírovať napríklad aj po zobrazení vo webovom prehliadači. Skutočná komplikácia nastáva pri pokuse o extrakciu hierarchicky štrukturovaných dát, čomu sa venuje aj táto práca. Komplikácia súvisí s nejasnou definíciou bielych medzier, nakoľko nie je presne známe, čo konkrétne sa snažia oddeliť alebo či sa jedná o tabuľku.

Predstavme si napríklad tradičnú tabuľku s bunkami oddelenými plnou čiarou. Každá bunka v sebe uchováva dáta, ktorých význam je okrem hodnoty definovaný aj ich polo-

hou v konkrétnom riadku a stĺpci. Inými slovami, na polohe bunky záleží a pre správnu interpretáciu musí byť po prevode na iný formát zachovaná.

Potreba získavania dát z PDF súborov je rozšírená a dobre známa. Je teda prirodzené, že existujú nástroje a postupy, ktoré sa tento proces snažia čo najviac uľahčiť. Táto práca skúma a pracuje s postupmi, ktoré sú zamerané na extrakciu dát z digitálnych neoskenovaných dokumentov a konkrétne na postupy implementované v programovacom jazyku Python.

1.2 Spracovanie PDF súborov

Táto práca je zameraná na spracovanie tabuľkových dát v prenosnom dokumentovom formáte (PDF). Dokumenty tohto typu sú častým výstupom dát z rôznych meracích zariadení, obsahujú teda dáta, ktoré má zmysel ďalej analyzovať zo štatistického hľadiska.

V práci sa kladie dôraz najmä na spracovanie dopredu definovanej množiny typov vstupných súborov, pri ktorých vieme vždy predpokladať tvar tabuľky, ktorá sa v nich nachádza.

1.2.1 PDF formát

Táto podkapitola venujúca sa definícii súborov PDF bola prevzatá z [1]. Tento súborový formát vyvinula v roku 1993 spoločnosť Adobe Systems. Skratka PDF je odvodená z anglického Portable Document Format. Používa sa na ukladanie dokumentov, ktoré môžu následne byť bez zmien a v rovnakej forme zobrazené nezávisle od softvéru, hardvéru alebo operačného systému na ktorom boli vytvorené. Súbory typu PDF môžu obsahovať text, obrázky, tabuľky ale taktiež aj interaktívne formuláre, videá, animácie a podobne. Primárnym účelom formátu je dosiahnutie uniformity, teda sa snaží zabezpečiť, aby sa dokumenty na všetkých zariadeniach zobrazovali rovnako.

Na tento formát existujú voľne dostupné prehliadače na mnoho platforiem. Je možné ich prehliadať aj priamo v internetovom prehliadači, čo je skutočnosť využitá aj pri tvorbe tejto práce, konkrétne pri zaistení správnej extrakcie tabuľkových dát. Najznámejší je prehliadač spoločnosti Adobe, Acrobat Reader, ktorý ponúka zadarmo.

Súbor ADOBE PDF je členený na niekoľko súvislých blokov. Rozmery objektov sú udané v dvoch dimenziách ako súradnice (x, y). Formát PDF je kódovaný pomocou ASCII-85 kódovania. Pre podporu znakov národných abecied sa miesto znaku použije príslušný číselný kód.

Tento formát podporuje rôzne typy šifrovania a ochrany obsahu (napríklad proti kopírovaniu textu a obsahu, proti vytlačeniu, upravovaniu, vkladaniu poznámok a podobne) ako aj rôzne typy kompresií obrázkov.

1.2.2 Súradnicový systém dokumentov PDF

Súradnicový systém PDF súborov sa nazýva *User Space*. Jedná sa o plochý dvojrozmerný priestor, ktorý vystihuje fyzické parametre tlačeného papierového dokumentu. Jednotkou tohto priestoru sú body (*points*), pričom rozloženie týchto bodov je približne 28 bodov/cm (alebo 72 bodov/palec). [15]

Počiatok súradnicovej sústavy, bod (0,0), sa nachádza v ľavom dolnom rohu strany, narozdiel od grafických prostredí ponúkaných jazykmi Python alebo JavaScript, čo je potrebné korigovať pri pokusoch o získavanie súradníc napríklad po kliknutí myšou.

Body dokumentu majú x-ové a y-ové súradnice, pričom bod je štandardne definovaný ako *bod = (x, y)*. Súradnice x rastú zľava doprava (vertikálne), maximálna možná súradnica (pri štandardnom dokumente o veľkosti Media Box – A4) je 612. Súradnice y rastú zdola hore (horizontálne) a maximálna veľkosť je štandardne 792.

Poloha slov alebo ich častí je v dokumente definovaná štvoricou bodov, ktorá vytvára štvorsten, ktoré dané slovo ohraničuje.

1.3 Základné prístupy k získaniu dát z PDF

Napriek možnej prvotnej predstave, že získavanie dát z PDF dokumentov je jednoduché, nie je táto predstava úplne pravdivá. Tento proces sa nezaobíde bez svojich komplikácií. Pri práci s ostatnými známymi typmi dokumentov (napríklad DOC, XLS či CSV) sa používatelia môžu spoľahnúť na to, že dáta, ktoré vidia, vedia v rovnakom nezmenenom formáte premiestniť do iných súborov. Následne používatelia očakávajú túto skutočnosť aj od formátu PDF, kde čo i len jednoduché kopírovanie veľakrát neprinesie požadované výsledky.

Z používateľského pohľadu, neštandardné správanie PDF súborov môžeme pripísať faktu, že používatelia často pracujú s editovateľnými formátmi súborov, avšak PDF slúži len na prezeranie. Dáta uložené v tomto formáte nemusia byť exaktne logicky štrukturované, pretože sa neberie ohľad na možnosť ich dodatočného mazania alebo preskupovania. Jedinou úlohou dát vo formáte PDF je vyzerať rovnako ako vyzerali po exporte na zariadení ich autora, nakoľko súbory tohto typu sú primárne určené na zdieľanie a tlač.

Základné techniky, ktoré môžeme k získaniu dát použiť a ich úspešnosť je nasledujúca:

1. Jednoduché kopírovanie a vkladanie
   Dobre známe kopírovanie a vkladanie pomocou kláves CTRL-C, CTRL-V je veľmi intuitívnou cestou k získaniu dát z PDF súboru. Používateľ vyberie oblasť, ktorú chce preniesť, prekopíruje ju a vloží do iného (editovateľného) typu súboru. Tento prístup sa oplatí pri zväčša textových dátach, ale môže pri ňom dôjsť k neželaným zmenám spôsobeným nezhodami v kódovaní znakov národných abecied. Dáta treba po prekopírovaní skontrolovať. Neprekonateľnú prekážku pre tento prístup tvoria tabuľky, z ktorých vieme kopírovaním získať len text, bez zachovania ich štruktúry. Touto problematikou sa bližšie zaoberá podkapitola 1.5.

2. Ručné získavanie dát
   Jedným z riešení ako získať dáta, pokiaľ kopírovanie spôsobuje neúnosné množstvo chýb, je získať ich ručne, teda otvorením PDF dokumentu používateľom, ktorý je dané dáta schopný prečítať a následne štylizovať do želaného formátu tak, aby odpovedali stanoveným požiadavkám. Nevýhodou je, samozrejme, vysoká časová náročnosť v prípade veľkého počtu dokumentov potrebných na takúto úpravu, ale aj finančné náklady, ktoré sú s takouto extrakciou spojené. Faktom ale zostáva, že manuálna extrakcia je dodnes veľmi využívaná, čo je potvrdené aj počtom ponúk na externé získavanie týchto údajov pracovníkmi v oblasti *freelance*.

3. PDF konvertory
   Konvertory, ktoré ponúkajú prevod súborov z PDF do niekoľkých ďalších formátov sú ďalšou z jednoduchších možností ako problém extrakcie dát z PDF vyriešiť. Používatelia, ktorým sa samotné jednoduché kopírovanie nepodarí sa veľmi pravdepodobne rozhodnú riešenie vyhľadať na Internete, kde po zadaní ich problému dostanú rozsiahlu ponuku online konvertorov, ktoré ponúkajú, že ich problém vyriešia. Pri

tomto spôsobe sa naskytuje zásadná nevýhoda týkajúca sa neistej bezpečnosti, keďže na spracovanie PDF dokumentu týmto online nástrojom je potrebné jeho nahratie na server sprostredkovateľa. Je teda na zvážení používateľa, či výsledná kvalita a rýchlosť spracovania, ktorá môže byť pri každom nástroji odlišná, stojí za možné bezpečnostné riziko úniku informácii z dokumentov, ktoré chce spracovávať.

4. Nástroje na extrakciu tabuliek
   Žiaden z vyššie spomínaných nástrojov zatiaľ nebral od úvahy možnosť, že v texte sa môžu nachádzať aj iné typy dát, napríklad štruktúrované. Zatiaľ čo doterajšie prístupy mali za úlohu presne rozoznať slová, oddeliť ich a prípadne zachovať správne kódovanie medzinárodných abecied, pri extrakcii tabuliek naberajú zvýšenú dôležitosť rozmiestnenia jednotlivých informácii v rámci rozloženia strany. Nástroje na extrakciu tabuliek z dokumentov typu PDF sa zameriavajú na rozpoznanie štruktúrovanej skupiny dát, pričom toto rozpoznanie realizujú:

   - automaticky, hádaním miesta v dokumente, ktoré najviac pripomína tabuľku,
   - zameraním sa na používateľom definovanú oblasť, v ktorej by sa tabuľka mala nachádzať.

   Zadanie konkrétnej oblasti tabuľky prispieva k optimálnejšiemu výsledku, keďže sa skracuje doba prehľadávania nepotrebných úsekov v dokumente a znižuje sa výskyt neželaných chýb.

Použitie niektorej vyššie uvedenej techniky na extrakciu dát z PDF závisí od konkrétneho typu spracovávaného dokumentu. Rôzne dokumenty vyžadujú rôzne prístupy, od najľahších a najintuitívnejších až po také, ktoré vyžadujú manuálny ručný prepis. Nepochybne existuje celý trh ponúkaných možností, pri ktorých sa oplatí brať do úvahy rýchlosť spracovania, finančnú náročnosť, objem dokumentov, exaktnosť výsledkov, ale v neposlednom rade aj bezpečnosť spracovávaných údajov.

1.4 Porovnanie nástrojov na extrakciu dát z PDF v prostredí Python

Doteraz boli spomínané všeobecné techniky, ktoré je možné použiť aj priamo z pohodlia internetového prehliadača. V nasledujúcej časti sa zameriame na konkrétne, už existujúce riešenia, ktoré sú dostupné v jazyku Python. Nasledujúca časť tejto podkapitoly bola prevzatá z [4]. Tieto riešenia sú ponúkané nasledujúcimi knižnicami:

1. PDFMiner
   Táto knižnica je zameraná na extrakciu významných informácií z PDF dokumentov. Na rozdiel od ostatných nástrojov sa zameriava len na získanie a analýzu daných dát.

   Výhody:

   - získava presnú polohu textu a ďalších objektov,
   - získava informácie o fontoch, farbách a ďalších vlastnostiach dokumentu,
   - rozdeľuje dokument na časti,
   - analyzuje a konvertuje do iných formátov.

Nevýhody:

- neponúka grafické užívateľské rozhranie,
- je nástrojom príkazového riadku,
- slabá dokumentácia.

2. PyPDF2
   Knižnica je celá napísaná v jazyku Python, dokáže rozdeľovať PDF súbory na časti, alebo ich naopak spájať do jedného súboru. Dokumenty dokáže tiež transformovať a získavať z nich dáta. Je schopná zabezpečiť dokument heslom, alebo špecifikovať vlastné dáta, prípadne možnosti zobrazenia. Súčasťou jej funkcií je aj zber metadát o dokumente.

   Výhody:

   - získanie metadát,
   - získanie textu,
   - funkcie zlučovania a oddeľovania PDF dokumentov.

   Nevýhody:

   - nižšia presnosť extrakcie dát v porovnaní s knižnicou PDFminer,
   - nie je schopná získať objekty typu obrázok, graf alebo iné,
   - vysoká robustnosť nástrojov s množstvom nastavení predstavuje komplikáciu pre začínajúcich používateľov.

3. Tabula-py
   Vychádza z implementácie tabula-java, pôvodne v jazyku Java, ale prináša všetky potrebné nástroje do prostredia programovacieho jazyka Python. Knižnica je zameraná špecifiky na potreby extrakcie tabuliek zo súborov typu PDF. Ponúka možnosť konvertovať tabuľky, či už do významných dátových štruktúr vhodných na ďalšie spracovanie priamo v jazyku Python, alebo do všeobecne zobraziteľných a používaných dátových typov JSON, TSV alebo CSV.
   Výhody:

   - zameranie sa na extrakciu tabuliek,
   - viacero podporovaných výstupných formátov,
   - vhodný formát výstupu na ďalšie spracovanie.

   Nevýhody:

   - podporuje len vyhľadávanie v digitálnych (neoskenovaných) PDF dokumentoch, ktoré umožňujú zvýraznenie textu,
   - nižšia presnosť extrakcie pri komplexnejších PDF dokumentoch.

4. PDFQuery
   Ľahká nadstavba nad knihovňami *pyquery*, *lxl* a *PDFminer*, ktorá ich spája a kombinuje s cieľom zaistiť export údajov, ktorý je možný už po napísaní pár riadkov kódu. Zjednodušuje a zapúzdruje prácu s inak komplexnými nástrojmi.
   Výhody:

- je schopná transformovať PDF dokument do stromovej štruktúry, v ktorej je následne možné sa orientovať a pohybovať pomocou webu blízkych selektorov typu *jQuery*,
- dokáže vyhľadávať niektoré elementy podľa pozície v dokumente.

Nevýhody:

- prvotné spustenie je časovo náročné, nakoľko je počet na začiatok vyhodnocovaných elementov na strane PDF dokumentu veľký,
- využitie hľadania elementu na pozícii je možné použiť len ak táto pozícia je na rôznych dokumentoch úplne rovnaká.

1.5 Techniky rozpoznania tabuliek

Veľkou výzvou pri exporte tabuliek z PDF súborov je fakt, že v kódovaní PDF tabuľky nenájdeme znak pre biele medzery. Slová sú jednoducho posunuté o kúsok doprava. Z pohľadu čitateľa je rozdiel nebadateľný. Avšak, pri spätnej snahe o rozšifrovanie sémantického významu jednotlivých celkov dokumentu je absencia konkrétneho oddeľovacieho znaku značnou komplikáciou. [2]

Nasledujúce rozdelenie algoritmov bolo prevzaté z [9]. Väčšina nástrojov na extrakciu tabuliek používa dva základné prístupy na získanie tabuľky z PDF súboru, ktoré sú postavené na nasledujúcich algoritmoch:

A) Technika *Stream*
Stream je technika používaná na rozbor a lokalizáciu medzier a prázdnych miest v dokumente, ktoré oddeľujú jednotlivé slová alebo iné ucelené časti informácií. *Stream* pracuje s náhodnosťou a tipovaním, pretože sa snaží odhadnúť tieto miesta na základe ich veľkosti a frekvencie výskytu.

Túto techniku sa oplatí použiť pre tabuľky, ktoré nemajú jednotlivé bunky graficky oddelené čiarami.

B) Technika *Lattice*
Lattice je ďalšia používaná technika, ktorá sa už nespolieha na tipovanie a odhady, ale hľadá vodorovné a zvislé čiary patriace tabuľkám. Táto technika funguje len na viditeľné čiary, ktoré oddeľujú jednotlivé bunky. Algoritmus používaný touto technikou pozostáva z nasledujúcich krokov:

1. Prevod PDF dokumentu na obrázok pomocou interpreta PDF súborov (*GhostScript*).
2. Aplikácia morfologických transformácií na získanie vertikálnych a horizontálnych čiar v obrázku.
3. Detekcia priesečníkov čiar pomocou kombinácie logického *AND* aplikovaného na čiarové segmenty získané v predchádzajúcom kroku a merania hustoty a intenzity pixelov tabuľky.
4. Detekcia okrajov tabuľky pomocou logického *OR* aplikovaného na čiarové segmenty (z kroku číslo 2) a hustoty ich pixelov.
5. Rozdelené alebo zlúčené bunky sú vyhľadávané pomocou priesečníkov čiar a ich segmentov.

6. Detegované segmenty čiar a okraje tabuľky sú následne škálované a namapované späť do PDF dokumentu, nakoľko mierka a dimenzie obrázku sa od pôvodného dokumentu môžu líšiť.
7. Po spätnom umiestnení čiarovej mriežky na príslušné súradnice *(x, y)* sú konkrétne slová, ktoré patria do jednotlivých buniek, uložené do vhodnej reprezentácie. Táto reprezentácia má podobu hierarchickej dátovej štruktúry, ktorá zodpovedá pôvodnému rozloženiu.

Kapitola 2

Štatistická regulácia procesov

Táto kapitola poskytuje úvod do problematiky štatistickej analýzy a štatistickej regulácie procesov. Je venovaná predstaveniu základných matematických princípov a pravidiel, ktoré sú neskôr využívané pri implementácii výslednej aplikácie a proces ich využitia je bližšie opísaný v Kapitole 3. V tejto kapitole sú predstavené všetky nástroje, ktoré budú potrebné k spracovaniu dát, ktoré už boli extrahované z PDF súborov.

Štatistická regulácia procesov je motivovaná zaistením bezprostrednej a pravidelnej kontroly kvality vykonávaného procesu, pričom toto hodnotenie sa realizuje pomocou dostupných matematických a štatistických hodnotení kvality. Umožňuje identifikovať a zamedziť nedostatočnú výslednú kvalitu vykonávaného procesu. [7]

Hlavnými cieľmi štatistickej regulácie procesov sú:

- dosiahnutie štatisticky stabilného stavu procesu (proces vtedy môžeme považovať za zvládnutý),
- udržovanie procesu na požadovanej stabilnej úrovni,
- rozlišovanie medzi náhodnými a zvláštnymi príčinami variability procesu,
- čo najrýchlejší zásah do procesu, v ktorom pôsobia zvláštne príčiny,
- identifikácia nepriaznivých vplyvov [7].

2.1 Význam variability v štatistickej regulácii

Dôležitým parametrom, ktorý pri štatistickej regulácii procesov sledujeme, je variabilita. Obsah tejto podkapitoly bol prevzatý z [7].

Čím je variabilita sledovaného procesu nižšia, tým stabilnejšie kvalitatívne výsledky proces produkuje, čo považujeme za kladný jav, nakoľko sa znižuje pravdepodobnosť výskytu produktov nesplňujúcich požadovanú kvalitu. Dosiahnutím nízkej variability procesu získavame úsporu času a nákladov potrebných na kontrolu, skúšanie a opravu výsledných produktov procesu.

Variabilitu detegujeme a monitorujeme v čase a odhaľujeme pomocou nej meniace sa parametre procesu.

Variabilitu pri štatistickej regulácii procesov delíme na základe dvoch kritérií:

1. Podľa toho, čo ju vyvolá:

   (a) Náhodné (prirodzené) príčiny

   Tvoria veľkú množinu neidentifikovateľných príčin, z ktorých každá malou mierou prispieva k celkovej variabilite. Odstránenie týchto príčin je často komplikované a vo väčšine prípadov nie je možné ich všetky eliminovať.

   Príklady náhodných príčin: chvenie stroja, teplota a vlhkosť ovzdušia, nerovnomernosť zloženia materiálu.

   (b) Zvláštne príčiny

   Tieto príčiny za bežných podmienok variabilitu neovplyvňujú. Avšak, ak k nim dôjde, vyvolávajú neprirodzené odchýlky v meraných hodnotách a sledovanej variabilite. Tento ich potenciálne veľký vplyv v kombinácii s ich obvykle jednoduchým odstránením odôvodňuje prečo sa pri štatistickej regulácii procesov zameriavame práve na ich elimináciu.

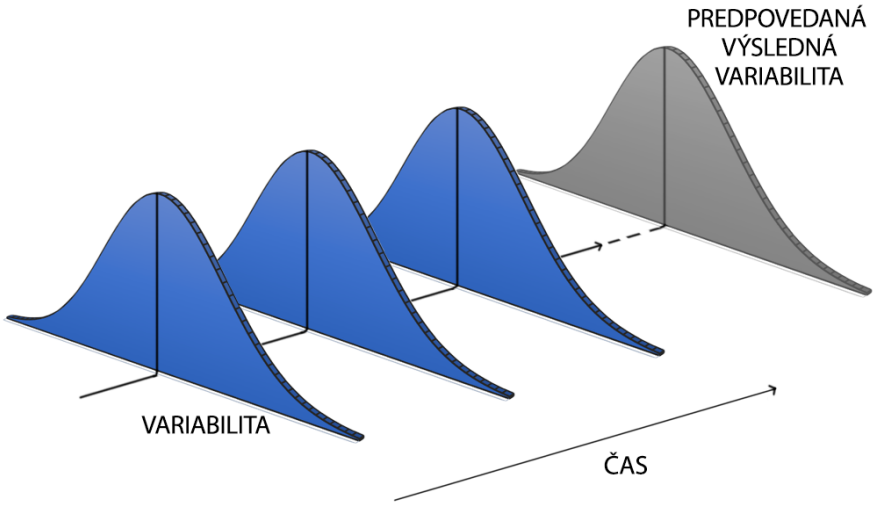
   Príklady zvláštnych príčin: poškodenie stroja, zmena nastavení stroja, nezaškolená obsluha.

   Zvláštne príčiny vieme ďalej rozdeliť podľa doby trvania počas ktorej sledovaný proces ovplyvňujú na:

   - sporadické príčiny,
   - pretrvávajúce príčiny.

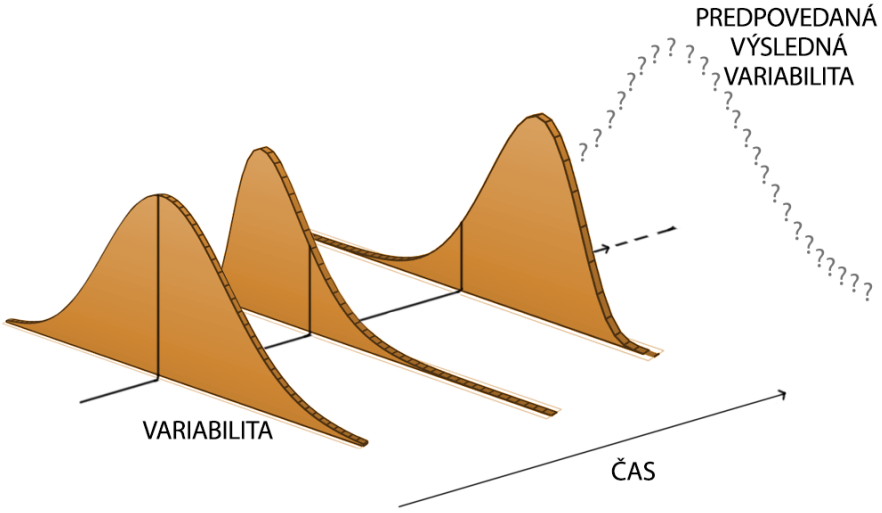
2. Podľa vplyvu zvláštnych príčin na variabilitu:

   (a) Štatisticky zvládnuté procesy (nepôsobia v nich zvláštne príčiny, iba náhodné)



Obr. 2.1: Predpoveď variability pri štatisticky zvládnutých procesoch.

   (b) Štatisticky nezvládnuté procesy (pôsobia v nich aj zvláštne príčiny)



Obr. 2.2: Predpoveď variability pri štatisticky nezvládnutých procesoch.

2.2 Regulačné diagramy

Regulačný diagram je základným grafickým nástrojom štatistickej regulácie procesov. S jeho pomocou sme schopní zachytávať zmeny variability procesu v čase a overovať štatistické hypotézy. Všeobecne je použiteľný všade tam, kde zaznamenávame informácie o kvalite v priebehu času, avšak najmä pri opakovaných procesoch, kde popri relatívne stabilných podmienkach výroby pôsobia aj ďalšie vplyvy.

2.2.1 Rozdelenie regulačných diagramov

Nasledujúce rozdelenie bolo prevzaté z [14].
Regulačné diagramy delíme podľa rôznych kritérií:

1. Podľa charakteru regulovanej veličiny:
   (a) Regulácia meraním – výberová charakteristika spojitého typu
   (b) Regulácia porovnávaním – výberová charakteristika diskrétneho typu
2. Podľa počtu regulovaných veličín:
   (a) Jedna veličina (Schewhartove diagramy)
   (b) Viacero veličín (Hotellingove diagramy)
3. Podľa ukladania predchádzajúcich hodnôt:
   (a) Bez pamäte – pri výpočte aktuálnej hodnoty regulovanej veličiny sa neuvažujú predchádzajúce namerané hodnoty (Shewhartove diagramy)
   (b) S pamäťou – pri výpočte aktuálnej hodnoty regulovanej veličiny sa uvažujú aj predchádzajúce namerané hodnoty (diagram kumulatívnych súčtov – CUSUM, diagram exponenciálne váženého kĺzavého priemeru – EWMA)

2.3 Fázy štatistickej regulácie procesov

Nasledujúce fázy boli prevzaté z [16].

1. Prípravná fáza
    - stanovenie cieľa regulácie
    - nastavenie parametrov procesu a znakov kvality
    - stanovenie kontrolných miest s dôrazom na realizáciu kontroly, čo najskôr po vzniku prípadnej odchýlky s cieľom zamedzenia nadbytočných nákladov na opravy a odpady
    - zvolenie vhodnej dĺžky intervalu kontroly
    - správne vytvorenie logických podskupín, v rámci ktorých sa predpokladá len pôsobenie náhodných príčin
2. Fáza zabezpečenia stavu štatistickej stability procesu
    - rozpoznanie pôsobenia zvláštnych príčin
    - zníženie účinkov identifikovaných zvláštnych príčin na najnižšiu možnú úroveň
    - vytvorenie podmienok, ktoré zamedzia opakovanému účinku identifikovaných zvláštnych príčin
    - stanovenie regulačných hraníc regulačných diagramov
3. Fáza analýzy a zabezpečenia spôsobilosti procesu
    - analýza spôsobilosti procesu k dosiahnutiu definovaných požiadaviek na základe informácií získaných v predchádzajúcom kroku
4. Fáza vlastnej štatistickej regulácie procesu
    - udržovanie procesu v štatisticky zvládnutom stave pomocou regulačného diagramu a regulačných hraníc stanovených v 2. fáze (viď. fáza 2).
    - odhaľovanie a zachytávanie porúch narušujúcich stabilitu procesu

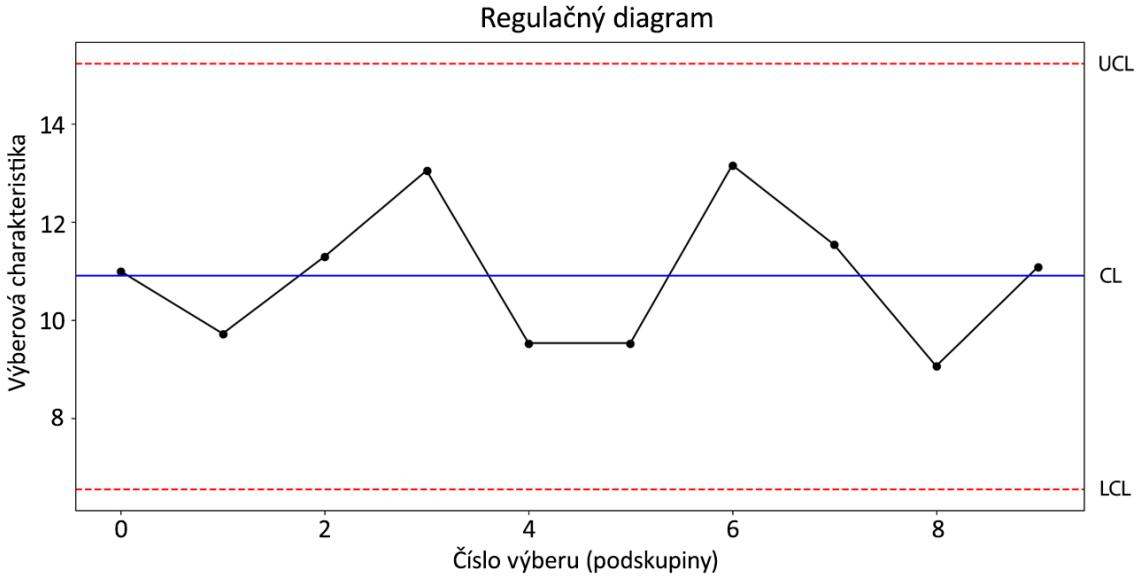
2.4 Základná charakteristika regulačného diagramu

Významnou prednosťou regulačného diagramu je jeho schopnosť signalizácie zvláštnej príčiny, keď k nej dôjde, pričom je schopný vyvarovať sa nepotrebnému falošnému signálu, pokiaľ k žiadnej dôležitej sledovanej príčine nedošlo.

Nevyhnutnosťou pri zostrojovaní regulačných diagramov je (napríklad na rozdiel od zostrojovania histogramov) zachovanie poradia získavaných údajov v čase.
Regulačný diagram predstavuje zobrazenie dát v dvojrozmernom súradnicovom systéme, kde:

- os x je časovou osou, sú do nej zaznamenávané poradové čísla časových okamihov, kedy boli realizované jednotlivé výbery (alebo poradové čísla výberov), pričom tieto výbery majú charakter logickej podskupiny,
- os y je osou dát/hodnôt, sledujeme na nej hodnoty zvolenej testovej štatistiky.

Logická podskupina predstavuje prevedenie výberu jednotiek produktu takým spôsobom, aby mala v rámci tohto výberu šancu sa prejaviť len variabilita vyvolaná náhodnými príčinami. Ak začne na proces pôsobiť nejaká zvláštna príčina je šanca na to, že sa v rámci podskupiny prejaví minimálna. Naopak, medzi podskupinami je šanca maximálna, čo nám umožňuje modelovať podskupinu ako jeden bod na osi x a k nej prislúchajúcu hodnotu na osi y. Rozdiely medzi jednotlivými podskupinami, medzi ktorými nás zaujímajú odchýlky vo variabilite graficky sledujeme ako prípadnú prudkú zmenu hodnôt v grafe. [5]



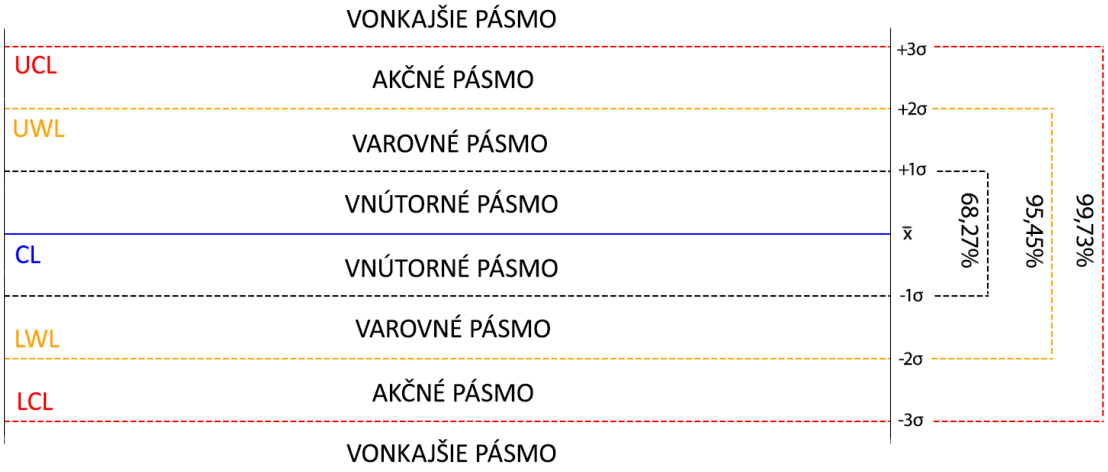
Obr. 2.3: Ukážka regulačného diagramu.

Priamky, ktoré sú v grafe vyznačené slúžia ako regulačné hranice a pomáhajú nám určiť, či je graficky zaznamenávaný proces štatisticky zvládnutý. Tieto priamky sú:

- LCL – dolná regulačná hranica
- UCL – horná regulačná hranica
- CL – centrálna priamka, ktorá odpovedá referenčnej hodnote a môžeme ju určiť rôznymi spôsobmi:
    1. Odhadom z hodnôt regulovanej veličiny získaných v podmienkach štatisticky zvládnutého procesu.
    2. Normálnou hodnotou, ak proces vieme zjednodušene popísať danou hodnotou.
    3. Na základe hodnoty získanej minulou skúsenosťou s daným procesom [16].

Regulačné hranice LCL a UCL sú stanovené staticky, pri regulačných diagramoch Shewhartovho typu sú väčšinou stanovené vo vzdialenosti 3σ (smerodajná odchýlka) danej výberovej charakteristiky. Vzdialenosť 3σ je vymeraná na obe strany od centrálnej priamky CL a pásmo takto vzniknuté (medzi LCL a UCL) obsahuje podľa pravidla tri sigma približne 99,7% všetkých hodnôt pri normálnom rozdelení pravdepodobnosti.

Toto pásmo pre nás vymedzuje pôsobenie len náhodných príčin, a naopak hodnota, ktorá sa nachádza mimo regulačných hraníc predstavuje pôsobenie zvláštnej príčiny.



Obr. 2.4: Členenie šírky regulačného diagramu pre aritmetický priemer.

V niektorých typoch diagramov sa môžu tiež objaviť ďalšie, takzvané výstražné hranice:

- UWL – horná výstražná hranica,
- LWL – dolná výstražná hranica.

Pásmo medzi hranicami UWL a LWL je užšie ako pásmo medzi ULC a LCL, najčastejšie berie do úvahy hodnoty $\pm 2\sigma$. Takto vzniknuté varovné pásmo upozorňuje na zvyšujúcu sa variabilitu.

2.4.1 Zistenie stability procesu

Obsah tejto podkapitoly bol prevzatý z [10].

Posúdenie priebehu regulačného diagramu prebieha analýzou funkčných hodnôt priliehajúcich k jednotlivým vybraným podskupinám. Cieľom je určiť štatistickú stabilitu procesu.

Pri vykonávaní analýzy regulačného diagramu je potrebné identifikovať špecifické zoskupenia bodov, na ktoré bude nutné reagovať.

Nežiadúce zoskupenia, ktoré by narušili stabilitu procesu vieme zhrnúť do ôsmich základných pravidiel (viď. tabuľka 2.1 a obrázky 2.5, 2.6 a 2.7).

Pravidlá 1 a 2 slúžia ako ukazovatele náhlych a veľkých zmien hodnoty sledovanej CL (najčastejšie strednej hodnoty). Zvláštne príčiny, ktoré tieto zmeny spôsobili, sú najčastejšie jednorázové.

Pravidlá 3 a 4 zachytávajú menšie zmeny prebiehajúce v dlhšom časovom období (toto obdobie je určite dlhšie ako pri pravidlách 1 a 2).

Pravidlo 5 odhaľuje tendenciu hodnôt procesu stúpať alebo klesať, čo vyvoláva tzv. trend. Trend môže byť spôsobený napríklad opotrebovaním nástroja alebo stroja.

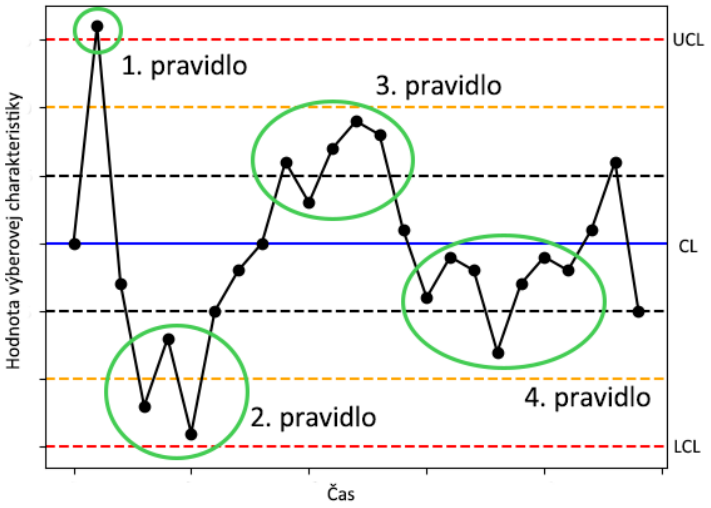
Pravidlo 6 upozorňuje na nadmernú variabilitu hodnôt, ktorá často nastáva v prípade prítomnosti viacerých sledovaných procesov, z ktorých každý je zaznačený samostatne. Riešením potreby sledovania viacerých procesov súčasne sú viacrozmerné pozorovania (viď. podkapitola 2.7).

Pravidlo 7 sa vyskytne v prípade snahy o zachytenie viacerých procesov v jednej spoločnej skupine, pri ktorej vytváraní vynikajú hodnoty veľmi blízke CL (odchýlky jednotlivých hodnôt sa vo výsledku vytrácajú). V regulačnom diagrame sa prejavuje veľkým zoskupením po sebe nasledujúcich bodov nachádzajúcich sa vo vnútornom pásme.

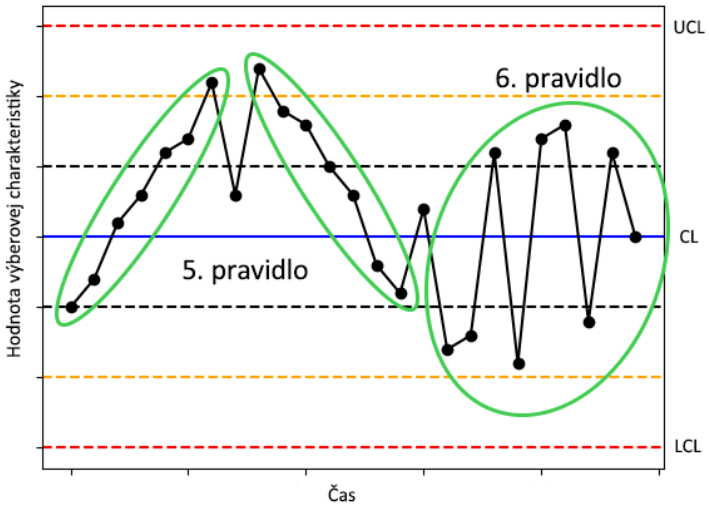
Pravidlo 8 je porušené najčastejšie kvôli nadmernej snahe o vonkajšiu kontrolu procesu, napríklad keď sa operátor stroja snaží dosahovať určité hodnoty. Ak sú dosahované hodnoty príliš vysoké, zmení nastavenia stroja a hodnoty klesnú. Naopak po prestavení stroja s nízkymi hodnotami sú ďalšie dosahované hodnoty vyššie. Tieto vonkajšie zásahy spôsobujú očividnú alternujúcu (kolísavú) tendenciu v grafe.

Číslo pravidla	**Názov pravidla**	**Hľadané zoskupenie**
1	Prekročenie hraníc	Jeden alebo viac bodov mimo regulačných hraníc
2	Akčné pásmo	2 z 3 po sebe idúcich bodov v akčnom pásme alebo za ním
3	Varovné pásmo	4 z 5 po sebe idúcich bodov vo varovnom pásme alebo za ním
4	Vnútorné pásmo	7 alebo viac po sebe idúcich bodov na jednej strane CL (vo vnútornom pásme alebo za ním)
5	Trend	7 po sebe idúcich bodov postupne klesajúcich alebo stúpajúcich
6	Nadmerná variabilita	8 po sebe idúcich bodov, z ktorých žiaden nie je vo vnútornom pásme
7	Nedostatočná variabilita	15 po sebe idúcich bodov vo vnútornom pásme
8	Alternujúce body	14 po sebe idúcich bodov, ktoré alternujú hore a dole

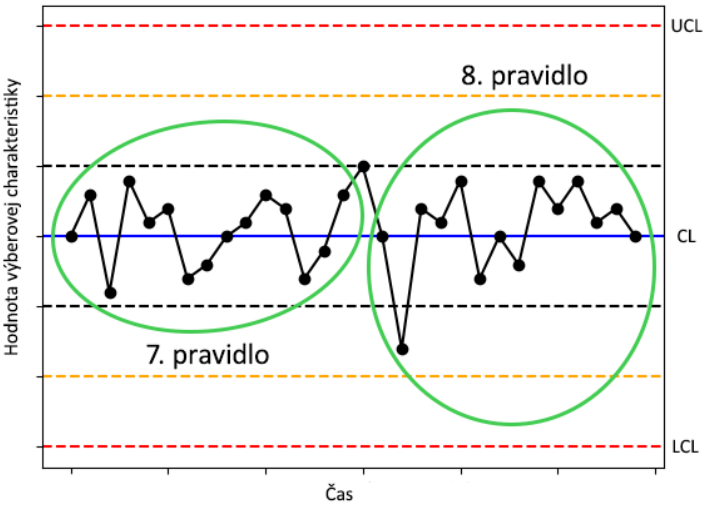
Tabuľka 2.1: Pravidlá pre zaistenie štatistickej stability procesu [10].



Obr. 2.5: Ukážka pravidiel 1, 2, 3 a 4.



Obr. 2.6: Ukážka pravidiel 5 a 6.



Obr. 2.7: Ukážka pravidiel 7 a 8.

2.4.2 Chyby a riziká ich výskytu

Pri analýze regulačného diagramu môže dôjsť k dvom typom chýb [7]:

1. Chyba prvého druhu (ISO 8258) – riziko falošného signálu

   Táto chyba vzniká, ak je proces štatisticky zvládnutý, ale napriek tomu niektorý z bodov padne do vonkajšieho pásma mimo kontrolné regulačné hranice UCL a LCL. Táto chyba môže mať za následok pokusy o odstránenie neexistujúceho problému.

Za predpokladu normálneho rozdelenia výberovej charakteristiky, s ktorou daný regulačný diagram pracuje vieme pravdepodobnosť tejto chyby vyčísliť ako:

$$\alpha = 1 - [\phi(L) - \phi(-L)] \tag{2.1}$$

Napríklad pre klasický Shewhartov regulačný diagram s hranicami 3σ $(L = 3)$ je: $\alpha = 1 - 0,9973 = 0.0027$

2. Chyba druhého druhu (ČSN 010265) – riziko chýbajúceho signálu

   Táto chyba vzniká, ak je proces štatisticky nezvládnutý, ale napriek tomu žiaden z jeho bodov neleží vo vonkajšom pásme a ani nie je súčasťou nenáhodného zoskupenia.

   Táto chyba vedie k neskorej identifikácii zvláštnych príčin a oddaľuje potrebný zásah do procesu.

   Pravdepodobnosť chyby druhého druhu vieme vyčísliť ako:

$$\beta = \phi(L - \delta\sqrt{n}) - \phi(-L - \delta\sqrt{n}) \tag{2.2}$$

   kde:

   - n je rozsah výberu (počet jednotiek),
   - ϕ je distribučná funkcia normovaného normálneho rozloženia pravdepodobnosti $N(\mu, \sigma^2/n) \sim \bar{x}$,
   - L je násobok smerodajnej odchýlky σ .

Obe tieto chyby je možné eliminovať sledovaním hodnôt v rámci ich rozloženia v pásmach, ako bolo spomínané vyššie.

2.5 Interpretácia regulačného diagramu

Po zostrojení regulačného diagramu je potrebné správne interpretovať výsledky, ktoré z neho vyplývajú, a to najmä to, či je štatisticky zvládnutý alebo nezvládnutý.

Rozhodnutie o štatistickej stabilite procesu:

- Proces považujeme za štatisticky zvládnutý, ak všetky jeho body ležia medzi dolnou a hornou regulačnou hranicou a nedošlo k porušeniu testovaných kritérií spomínaných vyššie. Takto zvládnutý proces nevyžaduje ďalší zásah a výsledok takto kontrolovaného procesu môžeme považovať za primerane vyhovujúci.

- Proces považujeme za štatisticky nezvládnutý, ak niektorý z jeho bodov leží mimo regulačných hraníc alebo porušuje aspoň jedno testovacie kritérium. Na nezvládnutý proces je potrebné čo najrýchlejšie reagovať, zistiť príčiny, ktoré ho spôsobili a ich odstránením v čo najkratšom čase opätovne dosiahnuť štatistickú stabilitu.

Výsledok	**Skutočnosť**	
	Neplatí H_0	**Platí** H_0
Zamietame H_0	Správne rozhodnutie Pravdepodobnosť $1-\beta$	Chyba 1. druhu Pravdepodobnosť α
Nezamietame H_0	Chyba 2. druhu Pravdepodobnosť β	Správne rozhodnutie Pravdepodobnosť $1-\alpha$

Tabuľka 2.2: Možné výsledky testovania hypotéz [8].

2.5.1 Priemerná dĺžka behu (ARL)

Obsah tejto podkapitoly bol prevzatý z [8].

ARL (*Average Run Length*) je priemerná dĺžka behu, ktorá sa používa k zhodnoteniu účinnosti regulačných diagramov. Predstavuje priemerný počet výberov (bodov) vedúcich k signálu, teda k tomu, aby bolo možné detektovať prekročenie regulačných hraníc.

Princíp spočíva v testovaní nulovej hypotézy:

$$H_0 : \textit{Sledovaný proces je pod kontrolou.}$$

Nulová hypotéza je testovaná oproti alternatívnej hypotéze:

$$H_1 : \textit{Sledovaný proces je mimo požadovaný stav.}$$

ARL_0 je definované pre nulovú hypotézu H_0 nasledovne:

$$ARL_0 = \frac{1}{\alpha} \tag{2.3}$$

Hodnota ARL_0 je nepriamo úmerná riziku falošného signálu α (čím je α vyššia, tým je ARL_0 nižšie). ARL_0 určuje počet bodov potrebných k tomu, aby došlo k falošnému signálu.

ARL_1 je definované pre alternatívnu hypotézu H_1 nasledovne [11]:

$$ARL_1 = \frac{1}{1-\beta} \tag{2.4}$$

ARL_1 určuje počet bodov potrebných k tomu, aby nastala chyba chýbajúceho signálu.

Výpočet hodnoty ARL sa vykonáva pomocou simulácie procesu a pri nastavovaní hodnôt regulačných hraníc sa dbá o to, aby bola hodnota ARL_0 čo najvyššia a hodnota ARL_1 čo najnižšia.

2.6 Klasické Shewhartove regulačné diagramy

Diagramy Shewhartovho typu sú známe od roku 1931, kedy ich princíp sformuloval W. A. Shewhart. Najčastejšie sa používajú pri procesoch, v ktorých je problematické odlíšiť kolísania sledovaných hodnôt vypovedajúcich o kvalite pod vplyvom náhodných a systematických príčin, a pri ktorých sa predpokladá regulovaná úroveň sledovanej premennej v istom časovom úseku a v istom pásme spoľahlivosti. [5]

Shewhartove regulačné diagramy regulujú len jednu veličinu, neumožňujú zoskupovať a sledovať viacero kvalitatívnych znakov súčasne v rámci jedného bodu diagramu.

2.6.1 Postup konštrukcie

Pri zostrojovaní regulačného diagramu Shewhartovho typu postupujeme nasledovne [16]:

1. Príprava procesných dát odpovedajúcich príslušnej časti procesu, ktorú chceme skúmať.
2. Odhad strednej hodnoty (aritmetického priemeru) a smerodajnej odchýlky z pripravených dát.
3. Overenie platnosti predpokladov štatistickej stability regulačného diagramu.
4. Zostrojenie regulačného diagramu s centrálnou líniou CL a regulačnými hranicami UCL a LCL.
5. Nanesenie dát do zostrojeného regulačného diagramu.
6. Sledovanie prípadného výskytu zvláštnych príčin, ktoré by sa prejavili nečakanou zmenou správania procesu (najčastejšie prekročením regulačných hraníc).
7. Pokus o odhalenie tzv. priraditeľnej príčiny, ktorá vyššie odhalenú zvláštnu príčinu vyvolala.

2.6.2 Rozšírenia klasických diagramov Shewhartovho typu

Tieto rozšírenia môžeme realizovať pre zlepšenie možností monitorovania stability:

- Pridanie robustnosti – tento proces je iteratívny a spočíva v nájdení a odstránení bodu, ktorý porušuje UCL a LCL hranice a opätovnom prepočítaní týchto hraníc, nakoľko boli týmto okrajovým bodom výrazne ovplyvnené. Robustné procedúry často využívajú mediány a mediánovú absolútnu odchýlku (MAD) namiesto strednej hodnoty a smerodajnej odchýlky.
- Zmena limitov – spomínané limity $\pm 3\sigma$ nie sú pevne dané, v praxi môžu byť upravené s cieľom dosiahnutia požadovanej rovnováhy v pravdepodobnostiach chýb prvého a druhého druhu.
- Zmena veľkosti výberu podskupiny (n) – zväčšenie veľkosti podskupiny vedie k citlivejšiemu detekčnému systému na odhaľovanie zmien v strednej hodnote. Regulačné hranice sú bližšie pri sebe, ale väčšia hodnota n vedie aj k predĺženiu doby potrebnej na odhalenie výkyvov, nakoľko v podskupine sa priemer získava z väčšej vzorky dát. [3]
- CUSUM – špeciálne regulačné diagramy, ktoré využívajú kumulatívne súčty k rýchlej detekcii malých a stredných zmien. Sú dostatočne citlivé na to, aby boli schopné detektovať zmeny spôsobené posunom strednej hodnoty. Táto citlivosť je dosiahnutá na úkor veľkosti výberu (CUSUM sa oplatí využiť pri malom počte výberov). [8]

2.7 Význam viacrozmerných pozorovaní

Doteraz sme sa zaoberali diagramami, ktoré sledujú jeden kvalitatívny znak v čase. V praxi však často ukazovateľom kvality nie je len jeden znak, ale ich skupina, pričom sa na výslednom podiele kvality podieľajú v rôznych množstvách. Je potrebné vedieť zaznamenať aj takéto dáta a na to nám slúžia diagramy pre viacrozmerné pozorovania.

Cieľom diagramov, ktoré takéto pozorovania využívajú je okrem odhaľovania zvláštnych príčin, ktoré spôsobujú zmeny v procesoch (čo bolo cieľom aj pri jednorozmernom pozorovaní), aj sledovanie kombinácií jednotlivých hodnôt (niektoré kombinácie so zvláštnymi hodnotami môžu byť prípustné, iné nie).

2.8 Hotellingove regulačné diagramy (viacrozmerné)

Obsah tejto kapitoly a vzorce v nej boli prevzaté z [7].

Hotellingov T^2 regulačný diagram je široko používaný nástroj pre monitorovanie viacerých navzájom súvisiacich charakteristík kvality v jednom diagrame. Vychádza zo štatistiky navrhnutej Hotellingom (1947), ktorá umožňuje prevod viacrozmerných pozorovaní usporiadaných vo forme vektorov, na hodnoty vzdialeností od stredných hodnôt. Tento prevod umožňuje grafické znázornenie, podobne ako pri jednorozmerných regulačných diagramoch.

Rozdiel medzi Hotellingovými diagramami a diagramami CUSUM a EWMA je v tom, že Hotelling nevyužíva celú históriu procesu, je preto menej citlivý na menšie zmeny vektoru strených hodnôt.

Cieľom tohto typu diagramu je zistiť, či nedošlo k zmene polohy vektora stredných hodnôt v dôsledku existencie zvláštnej príčiny. Vektor predstavuje zoskupenie p sledovaných znakov: namiesto jednotlivých sledovaných hodnôt $x_1, x_2, ..., x_n$ uvažujeme p-rozmerné vektory.

Pri Hotellingových diagramoch pracujeme s:

- p-rozmerný vektor výberových priemerov

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{2.5}$$

- vektor strených hodnôt (centroid) μ

$$\mu = \frac{1}{k}\sum_{j=1}^{k} \overline{x}_j \tag{2.6}$$

- výberová kovariančná matica typu $p \times p$

$$S = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})(x_i - \overline{x})^T \tag{2.7}$$

- kovariančná matica Σ

$$\Sigma = \frac{1}{k}\sum_{j=1}^{k} S_j \tag{2.8}$$

- hustota

$$f(x) = \frac{1}{\sqrt{(2\pi)^p\,|\Sigma|}}\, e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)} \tag{2.9}$$

2.8.1 Výpočet rozdelenia štatistiky T^2

Sledovanie viacerých kvalitatívnych znakov, je možné aj zostrojením jednorozmerných diagramov pre každý z nich. Nevýhodou takéhoto prístupu je, že strácame informácie o korelácii medzi jednotlivými znakmi.

Hotellingov diagram zohľadňuje aj koreláciu. Pri konštrukcii diagramu sa vychádza zo vzorca pre rozklad štatistiky T^2 [6]:

$$T^2 = n(\overline{x} - \mu)^T \Sigma^{-1} (\overline{x} - \mu) \tag{2.10}$$

kde:

- n je veľkosť podskupiny
- Σ^{-1} je inverzná kovariančná matica

2.8.2 Interpretácia Hotellingovho diagramu

Táto podkapitola a vzorce v nej spomínané boli prevzaté z [7].

Pre interpretáciu je potrebný správny výpočet regulačných hraníc, ktoré nám napomáhajú rozhodovať o stabilite sledovaných procesov.

Dolná regulačná hranica je vždy nulová a jej prekročenie do záporných hodnôt upozorňuje na prítomnosť zvláštnej príčiny.

Horná regulačná hranica sa dopočítava s využitím pravdepodobnosti a odvodzuje sa z rozdelenia štatistiky T^2. Rozlišujeme:

1. Určenie UCL v 1. etape – retrospektívne určenie na základe práve získaných dát

$$UCL = \frac{p(k-1)(n-1)}{kn-k-p+1} F_{1-\alpha}(p, kn-k-p+1) \tag{2.11}$$

2. Určenie UCL v 2. etape – určenie pre nasledujúcu reguláciu na základe už známych charakteristík rozdelenia

$$UCL = \frac{p(k+1)(n-1)}{kn-k-p+1} F_{1-\alpha}(p, kn-k-p+1) \tag{2.12}$$

Situácia s hľadaním možných príčin neželaných výkyvov hodnôt sa však komplikuje možnosťou existencie zvláštnej príčiny, v hociktorom zo sledovaných znakov. Po zachytení konkrétnej hodnoty, ktorá sa vymyká stanoveným regulačným hraniciam je potrebné zistiť, ktorého konkrétneho znaku sa problém týka. To vieme dosiahnuť okrem iných spôsobov aj použitím regulačných diagramov pre individuálne čiastkové znaky a rozšíriť regulačné medze tak, aby sa zmenšilo riziko falošného signálu a bola zachytená len podozrivá hodnota, dostatočne sa vymykajúca priemerom.

Rozklad a sledovanie jednotlivých hodnôt pre znaky individuálne je výhodné pri dvoch alebo troch znakoch, ale grafy sa dajú orientačne použiť aj pri väčšom počte sledovaných znakov.

2.9 Index spôsobilosti procesu

Index spôsobilosti počítame s cieľom zistiť mieru spoľahlivosti, s akou hodnoty sledovaného znaku zodpovedajú špecifikácii vychádzajúce s dopredu stanovených požiadaviek (napríklad zákazníka alebo normy).

Spôsobilosťou procesu chápeme schopnosť tohto procesu vyhovieť daným kritériám, pričom kritériá môžu byť stanovené ako jednostranné alebo obojstranné tolerančné hranice, ktoré nesmú byť prekročené žiadnou merateľnou hodnotou procesu. [7]

Pri regulačných diagramoch nám potrebné štatistické informácie boli podávané pre každú podskupinu, prehľadnosť teda pri vysokom počte meraní rapídne klesá. Index spôsobilosti na rozdiel od regulačných diagramov prináša komplexné informácie stručnejším spôsobom, konkrétne vedia byť zhrnuté do jedného či dvoch čísel, ktoré nám hovoria o celkovom splnení kritérií pre celý proces. Doplnkom k numerickým ukazovateľom môže byť aj histogram.

Spôsobilosť procesu môže byť vyjadrená ako priemerný podiel jednotiek neodpovedajúcich daným kritériám (tzv. nezhodných) alebo ako priemer počtu nezhôd na jednotku. [7]

2.9.1 Využitie indexu spôsobilosti v praxi

Táto podkapitola bola prevzatá z [7].

Hodnotenie spôsobilosti je v praxi veľmi opodstatnené, používa sa na overenie kvality výsledkov nejakého procesu (najčastejšie výrobného) a zákazníci môžu od dodávateľov vyžadovať konkrétne hodnoty. Dosiahnutie vysokej spôsobilosti procesu je žiadúci stav, ktorý môže viesť k úspore nákladov potrebných na výstupnú kontrolu alebo iné overenia kvality.

Index spôsobilosti procesu je ukazovateľ kvality využívaný pri procesoch, ktoré už považujeme za štatisticky zvládnuté. Vyhlásenie procesu za štatisticky zvládnutý nie je v súčastnosti chápané úplne jednoznačne. Niektoré procesy môžu pokojne vyhovovať koncovým požiadavkám napriek porušeniu konštantnej strednej hodnote, pričom toto porušenie ich v tradičnom kontexte robí staticky nezvládnutými.

Spôsobilosť procesu vieme vyhodnotiť aj pre viacrozmerné znaky kvality.

2.9.2 Prípustná a prirodzená variabilita

Pozorovanie prípustnej a prirodzenej variability slúži k správnej konštrukcii ukazovateľov. Merateľné znaky kvality mávajú predpisom stanovené [7]:

a) obojstranné tolerančné hranicu

b) len dolnú tolerančnú hranicu

c) len hornú tolerančnú hranicu

Obojstranné medze sa udávajú v tvare: $T \pm a$, kde T vyjadruje cieľovú alebo nominálnu hodnotu uprostred predpísanej tolerancie. Napríklad: $100 \pm 20\ psi$, čo predstavuje odporúčaný tlak v pneumatikách bicykla považuje za prípustné hodnoty v intervale od 80 do 120.

Pre predpísané tolerančné medze sa používajú označenia:

- LSL (Lower Specification Limit)
- USL (Upper Specification Limit)

Pokiaľ tieto hranice nie sú prekročené, môžeme danú jednotku z hľadiska uvažovaného meraného znaku považovať za zhodnú. Rozdiel hodnôt predpísaných tolerančných medzí predstavuje rozsah prípustnej variability skúmaného procesu.

2.9.3 Podiel nezhodných jednotiek

Tento podiel vyjadruje množstvo jednotiek s hodnotou znaku mimo predpísanú toleranciu. Podiel nezhodných jednotiek vieme určiť pomocou distribučnej funkcie $F(x)$ predpokladaného modelu rozloženia znaku kvality. Potom [7]:

- dolný podiel nezhodných jednotiek:

$$p_L = F(LSL) \tag{2.13}$$

- horný podiel nezhodných jednotiek:

$$p_U = F(USL) \tag{2.14}$$

Celkový podiel nezhodných jednotiek je:

$$p_t = p_L + p_U \tag{2.15}$$

2.9.4 Výpočty pre rôzne typy indexov spôsobilosti

Vzorce v tejto podkapitole boli prevzaté z [13].

Výpočet indexu spôsobilosti (Process Capability Index):

$$C_p = \frac{USL - LSL}{6\sigma} \tag{2.16}$$

C_p odhaduje, čo je proces schopný vyprodukovať, za predpokladu, že aritmetický priemer je v strede medzi kontrolnými regulačnými hranicami (CL je v strede medzi UCL a LCL) a výstup procesu odpovedá normálnemu rozloženiu pravdepodobnosti.

$$C_{p,lower} = \frac{\mu - LSL}{3\sigma} \tag{2.17}$$

$C_{p,lower}$ odhaduje spôsobilosť procesu pre špecifikáciu, ktorá vyžaduje len dolný limit (napríklad sila).

$$C_{p,upper} = \frac{USL - \mu}{3\sigma} \tag{2.18}$$

$C_{p,upper}$ odhaduje spôsobilosť procesu pre špecifikáciu, ktorá vyžaduje len horný limit (napríklad koncentrácia).

$$C_{pk} = \min\left[\frac{USL - \mu}{3\sigma}, \frac{\mu - LSL}{3\sigma}\right] \tag{2.19}$$

C_{pk} odhaduje, čo je proces schopný vyprodukovať, bez predpokladu, že aritmetický priemer je v strede medzi kontrolnými regulačnými hranicami. V prípade, že CL nie je vycentrovaná medzi UCL a LCL výpočet klasického C_p predpokladá vyššiu spôsobilosť ako je skutočná.

Ak $C_{pk} < 0$ tak CL sa nachádza mimo kontrolných regulačných hraníc (vo vonkajšom pásme).

$$C_{pm} = \frac{C_p}{\sqrt{1 + \left(\frac{\mu - T}{\sigma}\right)^2}} \tag{2.20}$$

Tagucgiho index spoľahlivosti – odhaduje spoľahlivosť procesu okolo cieľa T. C_{pm} je vždy väčšie ako nula.

$$C_{pkm} = \frac{C_{pk}}{\sqrt{1 + \left(\frac{\mu - T}{\sigma}\right)^2}} \tag{2.21}$$

Tagucgiho index spoľahlivosti – odhaduje spoľahlivosť procesu okolo cieľa T, pričom CL nemusí byť vycentorvaná.

2.9.5 Odporúčané hodnoty

Po vypočítaní indexu spôsobilosti získavame číslo, ktoré samo o sebe neposkytuje zhodnotenie spoľahlivosti procesu. Na takéto zhodnotenie potrebujeme výslednú hodnotu porovnať s odporúčanými minimálnymi indexmi spoľahlivosti. Proces môžeme vyhlásiť za štatisticky spôsobilý ak vypočítaná hodnota jeho indexu spôsobilosti neprekračuje príslušnú hodnotu (viď. tabuľka 2.3).

Prípad	**Odporúčaný minimálny index spôsobilosti pre obostranné špecifikácie**	**Odporúčaný minimálny index spôsobilosti pre jednostranné špecifikácie**
Existujúci proces	1,33	1,25
Nový proces	1,50	1,45
Bezpečnostný alebo kritický parameter pre existujúci proces	1,50	1,45
Bezpečnostný alebo kritický parameter pre nový proces	1,67	1,60
Proces kvality 6 sigma	2,00	2,00

Tabuľka 2.3: Odporúčané hodnoty indexu spôsobilosti [12].

Kapitola 3

Návrh výslednej webovej aplikácie

Po získaní základných znalostí z oblasti získania dát z PDF súborov a preskúmaní dostupných matematických nástrojov určených na štatistickú analýzu týchto dát v predchádzajúcich kapitolách nastal čas na pretavenie získaných informácií do praxe. Nasledujúca kapitola sa venuje návrhu výslednej aplikácie, ktorá tvorí výstup tejto práce. Budú v nej predstavené potrebné knihovne, návrh grafického užívateľského rozhrania a použité technológie.

Výsledná aplikácia je navrhnutá pre prácu vo webovom prehliadači, nakoľko je formát PDF webu blízky a v tomto prostredí jednoducho zobraziteľný. Riešenie pomocou webovej aplikácie navyše uľahčuje prehľadnosť. Používatelia sa nachádzajú v dobre známom a zaužívanom prostredí, je im tu blízke nahrávanie aj sťahovanie súborov. Prostredie Python zároveň poskytuje vhodné nástroje na vývoj webových aplikácií, čo umožňuje, že ovládanie aplikácie aj štatistické spravovanie dát sa môže odohrávať na jednom mieste.

3.1 Grafické používateľské rozhranie

Nakoľko riešenie je realizované pomocou výslednej aplikácie s grafickým rozhraním, s ktorou budú používatelia pracovať, je potrebné zakomponovať jej ovládacie prvky zmysluplne a prehľadne. Naskytuje sa potreba nájdenia správneho kompromisu medzi ponúknutím čo najväčšieho počtu nastavení a dostatočnej jednoduchosti a rýchlosti prechádzania medzi jednotlivými krokmi vedúcemu ku konečnému výsledku.
Aplikácia sa skladá z troch hlavných stránok:

- Strana výberu oblasti z PDF súboru (viď. obrázok 3.1)

  Základné menu v hornej časti ponúka jednoduchú možnosť na výber prednastavených oblastí, ktoré sa viažu k základným dokumentom, ktoré boli zohľadnené pri vypracovaní. Oblasť tabuľky môže byť vybraná aj vyznačením oblasti priamo do ukážky PDF dokumentu, ktorý sa nachádza nižšie. Oblasť sa môže naznačiť len na aktuálnej stránke. Následne je ponúknuté graficky odlíšené tlačidlo prechodu na ďalšiu časť spracovanie dát.

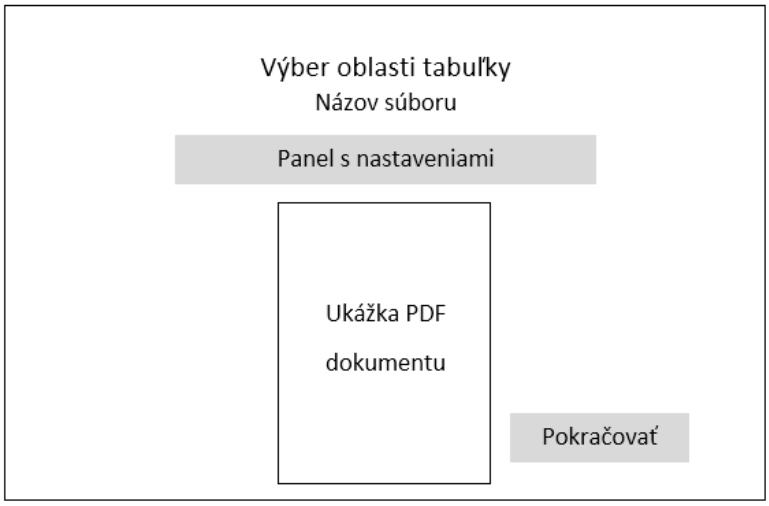
- Strana výberu dát z tabuľky (viď. obrázok 3.2)

  Tabuľky môžu obsahovať viacero riadkov a stĺpcov. Nie všetky dáta v nich môžu byť vhodné na štatistickú analýzu. Používateľ má v tomto kroku možnosť výberu tých dát, ktoré sú v skúmanom kontexte pre neho podstatné a ktoré chce zakomponovať do výslednej štatistickej analýzy. Ovládacie prvky na tejto stránke spočívajú z prepínača medzi riadkami a stĺpcami (dáta, ktoré spolu tvoria spoločnú logickú podskupinu

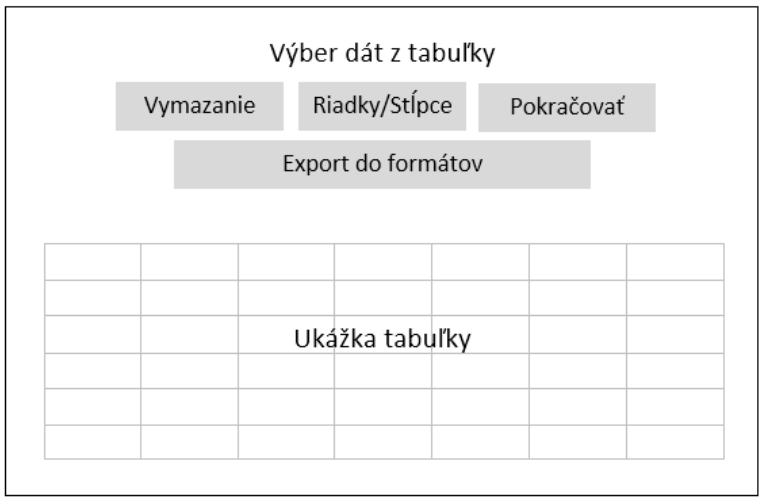
sú vzájomne usporiadané buď horizontálne alebo vertikálne) a tlačidlom na odstránenie výberu. Ďalej sa tu nachádzajú možnosti exportu získaných dát, v podobe ako ich užívateľ vidí do viacerých podporovaných formátov, čo predstavuje výhodu v prípade záujmu o využitie ako nástroja na digitalizáciu tabuliek bez potreby štatistickej analýzy.

- Strana zobrazenia regulačných diagramov a štatistík (viď. obrázok 3.3)

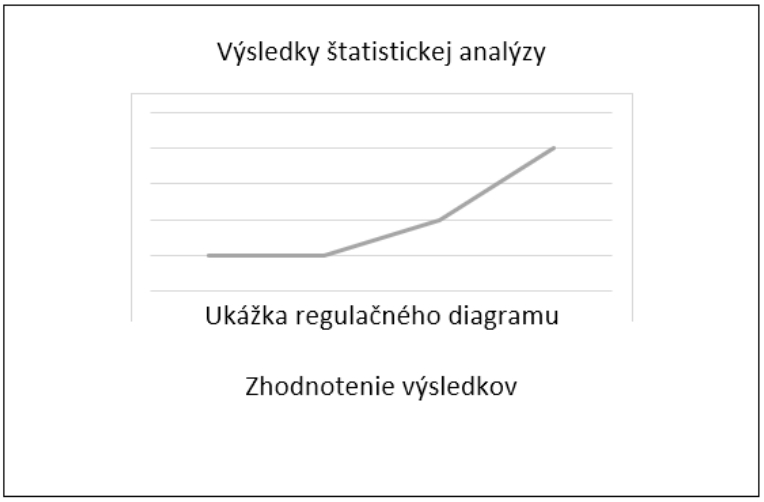
  Táto strana slúži na poskytnutie prehľadných výsledkov vykonanej analýzy. Nachádzajú sa na nej diagramy, pričom ich zobrazenie a rozloženie závisí na výbere predchádzajúcich nastavení. Základné zobrazenie (s diagramom Shewhartovho typu a jednou logickou podskupinou na analýzu) obsahuje jeden regulačný diagram v hornej časti.



Obr. 3.1: Základný návrh stránky výberu oblasti z PDF.



Obr. 3.2: Základný návrh stránky výberu dát z rozpoznanej tabuľky.



Obr. 3.3: Základný návrh stránky s výsledkami štatistickej analýzy.

3.1.1 Zobrazovanie upozornení

Pri vykonaní určitých nedovolených alebo potenciálne nechcených používateľských akcií je potrebné používateľa upozorniť zobrazením hlásenia. Takto ošetrené by mali byť hlavne nevratné akcie a akcie, ktoré by inak viedli k chybovým stavom aplikácie.

Používateľ môže byť napríklad upozornený na to, že nevybral konkrétnu oblasť tabuľky. Toto upozornenie mu ale dá možnosť pokračovať aj napriek tomu. Iným typom upozornenia je také, ktoré by nastalo v prípade, že nevyberie potrebné dáta z už rozpoznanej tabuľky na ďalšiu analýzu. Toto upozornenie mu nedovolí pokračovať, kým príslušné dáta nevyberie.

3.2 Využité technológie

3.2.1 Formáty uloženia tabuľky

Tabuľka sa počas extrakcie a následného spracovávania nachádza v niekoľkých formátoch:

- PDF – Portable Document Format
  Z tohto úvodného formátu tabuľku získavame. Tabuľka v PDF dokumente nie je počítačom priamo rozlíšiteľná pomocou analýzy zdrojového kódu dokumentu. Na jej nájdenie a extrakciu dát z nej môžeme použiť len nástroje, ktoré zohľadňujú štylizáciu a grafickú úpravu. Tabuľku teda vyslovene hľadajú.

- XML – Extensible Markup Language
  XML je značkovací jazyk, ktorý sa v implementácii knižnice *Tabula* používa na prevod PDF súboru do zoskupenia dát zrozumiteľných aj pre počítač. Vďaka tomuto jazyku je vytvorená zmysluplná hierarchia, ktorá odráža rozloženie v pôvodnom PDF dokumente. Samotný kód je tvorený pármi značiek, ktoré zastupujú rôzne časti dokumentu (napríklad `<page>` reprezentuje stranu, `<text>` reprezentuje časť textu). Každá značka obsahuje aj pozičné údaje v podobe súradníc.

- CSV – Comma-separated Values
  CSV je formát ukladania dát, ktorý jednotlivé hodnoty oddeľuje konkrétnym znakom

(najčastejšie čiarkou). Je jedným z výstupných formátov, ktoré ponúka knižnica Tabula a bol do výslednej implementácie vybraný pre jednoduchosť a priamočiarosť jeho prepojenia s ďalšou funkcionalitou potrebnou k rozpracovaniu dát v prostredí Python, konkrétne možnosť importovať dáta z CSV formátu priamo do DataFrame Pandas.

- Pandas DataFrame
  Výsledný formát uloženia tabuľky, z ktorého čerpáme konečné dáta priamo na výpočet a vykreslenie záverov štatistickej analýzy. Predstavuje dáta usporiadané do riadkov a stĺpcov a ponúka rôzne funkcie na ich vyberanie, usporadúvanie a ďalšie výpočty s nimi.

3.2.2 Použitie knižnice *Tabula*

Tabula je komplexný nástroj na extrakciu tabuliek z PDF dokumentov. Základný program je implementovaný v jazyku Java, avšak existuje aj zjednodušená verzia, ktorá vie byť importovaná ako knižnica tabula-py do prostredia Python.

Počas prvotného testovania v úvode vypracovania tejto práce, boli okrem knižnice *Tabula* otestované aj knižnice *Camelot* a *Excalibur*, pričom *Tabula* podávala na testovaných súboroch najrovnomernejšie a najpresnejšie výsledky a preto bola zvolená ako hlavný nástroj extrakcie do výslednej implementácie.

Užitočné nastavenia knižnice:

- Možnosť špecifikácie súradníc pozície tabuľky
  Tabula nemusí prehľadávať celý dokument a analyzovať v ňom všetky medzery alebo čiary (podľa vybraného algoritmu viď. 1.5), ale tieto techniky môže sústrediť len na dopredu špecifikovanú oblasť. Geometria definovaných oblastí je podobná ako pri PDF súboroch. Počiatok súradnicovej sústavy je v ľavom dolnom rohu. Maximálna hodnota súradnice x (maximálna šírka) je 596. Maximálna hodnota súradnice y (maximálna výška) je 842.

- Výber medzi technikami spracovania *Stream* a *Lattice*
  Základné technológie získavania dát z tabuliek boli špecifikované v časti 1.5. *Tabula* umožňuje medzi nimi vyberať, takže poskytuje ideálny nástroj ako na extrakciu čiarami neohraničených tabuliek, tak aj na tie, ktoré majú jednotlivé bunky graficky oddelené.

- Možnosť vyhľadania tabuľky na stránke pomocou parametru `guess`
  V prípade, že konkrétna oblasť tabuľky nie je známa alebo nie je používateľom definovaná vie *Tabula* vyhodnotiť celý dokument a odhadnúť, kde asi sa tabuľka nachádza. Následne ju spracuje rovnako, ako v iných prípadoch. Lepšie a rýchlejšie výsledky sú ale dosiahnuté v prípade výberu konkrétnej oblasti.

3.2.3 Reprezentácia dát v Pythone – Pandas DataFrame

Pandas DataFrame je výstupným formátom knižnice *Tablua*. Predstavuje najvyhovujúcejšiu reprezentáciu tabuľkových dát s prihliadnutím na optimálne spracovanie a ďalšie výpočtové operácie, ktoré je potrebné vykonať v ďalších krokoch.

Pandas Dataframe predstavuje dvojrozmernú, potenciálne heterogénnu, reprezentáciu tabuľkových dát s premenlivou dĺžkou. Môže obsahovať pomenované osi (hlavičky) tabuľky, ktoré pomenúvajú alebo číslujú stĺpce a riadky.

Výhody využitia knižnice Pandas:

- jednoduchá reprezentácia dát
- flexibilita a prispôsobivosť dát
- úsporné, krátke zápisy, ktoré umožňujú aj veľmi komplexné výpočty
- široké spektrum podporovaných funkcií
- efektivita spracovania dát o veľkých objemoch

3.2.4 Práca s matematickými funkciami knižnice *Numpy*

Ďalšou s využitých knižníc je matematická knižnica *Numpy*, ktorá prestavuje štandard pre prácu s numerickými dátami v prostredí programovacieho jazyka Python. Zo všestrannej ponuky jej funkcií sa do výslednej implementácie najlepšie hodili jej homogénne dvojrozmerné polia – *numpy.array*, ktoré slúžia na uchovanie matíc, s ktorými je následne možné vykonávať základné matematické operácie (najmä sčítanie, násobenie a inverzia). Táto funkcionalita bola využitá pri Hotellingových diagramoch, konkrétne pri výpočtoch rozložení T^2 pomocou kovariančnej matice.

3.3 Architektúra webovej aplikácie

Architektúra je postavená na návrhovom vzore *Model View Controller*, ktorý logicky oddeľuje časti podľa toho, na čo sa používajú.

View má na starosti zobrazovanie. Keďže sa jedná o webovú aplikáciu sú to použité tradičné, webovému prostrediu blízke, technológie. Základ je definovaný s využitím značkovacieho jazyka HTML, ktorý spolu so špeciálnymi prvkami získanými z ovládacieho modulu (*Flask*) určuje základnú stavbu zobrazovanej stránky, ktorá reprezentuje jeden krok v spracovaní a analýze PDF dokumentu a dát z neho získaných. Vzhľad je ďalej dotvorený s využitím kaskádových štýlov CSS, ktoré výslednému riešeniu dodávajú prehľadnosť a uniformitu. Vzhľadová logika a automatizované akcie ovládacích prvkov zabezpečuje jazyk JavaScript, ktorý je úzko prepojený z ovládacím modulom *Flask*.

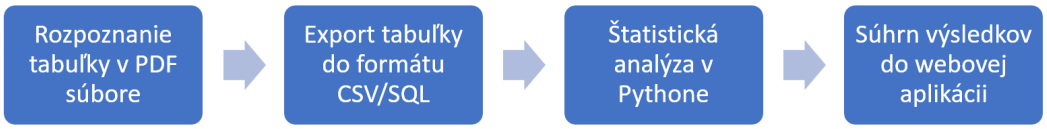
Model slúži k ovládaniu aplikácie. Je napísaný s využitím programovacieho jazyka Python, konkrétne jeho nadstavby na tvorenie webov – Flask. Ovládanie aplikácie spočíva v získaní vstupu od používateľa. Vstup, ktorý nás zaujíma je PDF súbor a ďalej jeho ohraničená oblasť, v ktorej očakávame výskyt tabuľky. Ovládacia logika je schopná získať lokálny súbor, s ktorým sa pracuje. Ďalej získava oblasť tabuľky a prevádza ju zo súradníczo súradníc získaných v jazyku JavaScript, ktoré sa viažu na plátno Canvas, do súradníc geometrie PDF dokumentu. Okrem toho umožňuje aj správnu reprezentáciu už získaných štruktúrovaných dát, ktoré odosiela zobrazovacej časti *View* na ukážku tabuľky v HTML. V neposlednom rade analyzuje vybrané dáta z tabuľky a vytvára z nich regulačné diagramy. Poskytuje štatistické zhodnotenie kvality sledovaného procesu na základe vyhodnotenia kritérií stanovených v časti 2.4.1 a vypočítava index spôsobilosti.

Controller prepája používateľské akcie s modelom. To, čo chce používateľ vykonať, je zasielané z *View* v podobe webovej aplikácie pomocou internetových protokolov *POST*, kde sa dáta ukladajú do formulárov a sú posielané priamo na vyhodnotenie do prostredia Python. Viac o komunikácii a výmene informácií medzi webovým prostredím a ovládaním je v podkapitole 4.2.2.

Kapitola 4

Podrobnosti implementácie

4.1 Postup získavania a spracovania dát



Obr. 4.1: Postup získania a spracovania dát.

Dáta, ktorých štatistickú spôsobilosť budeme neskôr hodnotiť, je najskôr potrebné rozpoznať v PDF súboroch, z ktorých sú následne exportované do formátu vhodného, či už na ich ďalšie spracovanie alebo štatistickú analýzu. Matematické operácie vyžadované pre štatistické úkony a následnú správnu číselnú a grafickú interpretáciu získaných dát sú vykonávané s využitím programovacieho jazyka Python. Konkrétne štatistické nástroje a ich matematické definície budú bližšie predstavené v Kapitole 2. Neskôr bude spomínaná ich implementácia a optimalizácia, s dôrazom na exaktné pretavenie numerických postupov do funkčného a použiteľného kódu, konkrétne v Kapitole 3.

Implementácia obsahuje všetky kroky potrebné k získaniu finálneho výsledku (od načítania dát až po ich finálne zobrazenie vo výslednej webovej aplikácii). Jednotlivým krokom implementácie sa budem venovať v nasledujúcej podkapitole.

Samotná webová aplikácia v sebe zahŕňa všetky prvky:

- načítanie PDF súboru,
- rozpoznanie tabuľky a hodnôt v nej,
- predpríprava dát do podoby vhodnej na ďalšie spracovanie,
- prevedenie štatistickej analýzy, výpočty hodnôt pre regulačné diagramy,
- grafické zobrazenie výsledkov a súhrnné zistenia.

Aplikácia je realizovaná pomocou webového rámca *Flask* pre Python. Umožňuje teda využiť všetky doteraz spomínané dátové štruktúry prostredia Python a prepája ich s načítaním a zobrazením z pohodlia webového prehliadača.

4.2 Aplikáciou vykonávané akcie a ich implementácia

Implementácia spočíva v sérii HTML stránok, z ktorých každá vedie ku konkrétnemu kroku v procese dodania, spracovania a vyhodnotenia dát. Stránky vykonávajú nasledujúce akcie:

1. Načítanie PDF súboru
   Aplikácia pracuje na lokálnom serveri. Na vstupe očakáva súbor typu PDF, ktorý si uloží do priečinku *static*, v ktorom sa nachádzajú všetky externé dáta, ku ktorým chceme aplikácii umožniť prístup. V prípade snahy o rozšírenie na vzdialený server by sa PDF súbor nahrával na dopredu stanovené umiestnenie na tomto serveri. Načítanie prebieha pomocou webového rozhrania s využitím odosielania formulára metódou *POST*.

2. Výber oblasti tabuľky

   - Automatický
     PDF súbor je zobrazený s využitým vstavanej funkcie webových prehliadačov na zobrazovanie dokumentov tohto typu. Používateľ má v tomto kroku možnosť výberu z predvolených profilov nastavení viažúcich sa na konkrétne typy tabuliek. Tieto profily boli zostavené s ohľadom na očakávané rozloženia tabuliek v dokumente a viažu sa na celé skupiny podobných dokumentov. Podobnými dokumentami môžeme chápať dokumenty, v ktorých je poloha a vizuálna reprezentácia tabuľky dostatočne zhodná na to, aby boli obe správne rozpoznané pri použití rovnakých nastavení.
   - Ručný
     Ďalšou možnosťou je, že užívateľ sám definuje oblasť tabuľky, ktorú má záujem spracovať. Tento výber je realizovaný kliknutím, potiahnutím a nakreslením útvaru pravouhlého štvoruholníka okolo želanej oblasti. Pre ďalšie spracovanie je potrebné získať informácie o tejto používateľom definovanej oblasti. Konkrétne potrebujeme zistiť súradnice jej krajných bodov. PDF súbor nám neponúka možnosť rozoznať, v ktorom mieste v ňom bolo myšou kliknuté. Z toho dôvodu nad sa nad ním vytvára priehľadný element *Canvas* z jazyka JavaScript, ktorý zrealizuje vykreslenie a zhromaždí súradnice. Tieto súradnice je ale potrebné dodatočne previesť do formátu PDF súradníc (viď. 1.2.2), pretože v tomto formáte pracuje aj knihovňa Tabula, ktorej sú zistené súradnice sprostredkované.

3. Výber konkrétnych dát na štatistickú analýzu
   Po výbere oblasti je používateľovi ponúknutý náhľad vybraných dát, z ktorých si vie vybrať len určitú časť. Zobrazené dáta tvoria logické podskupiny buď po riadkoch alebo po stĺpcoch. Výber týchto podskupín je v kompetencii používateľa rovnako ako aj voľba rozsahu sledovaných dát. Existuje možnosť jednorozmerného a viacrozmerného pozorovania. Používateľ vie v tabuľke vyznačiť určité množstvo podskupín. S ohľadom na prehľadnosť a celkovú použiteľnosť výsledkov nemá zmyesl uvažovať nad výberom viac ako štyroch podskupín naraz.

4. Grafický výstup štatistickej analýzy

   - Pre jednorozmerné pozorovanie
     Tento výstup zahŕňa len diagramy Shewhartovho typu, je výpočtovo menej

náročný a vie zobraziť niekoľko individuálne štatisticky posudzovaných grafov a hodnôt, pričom prípadná súvislosť medzi vybranými podskupinami je zanedbateľná a každá je vyhodnocovaná samostatne.

- Pre viacrozmerné pozorovanie
  Pre procesy s viacerými hodnotami ovplyvňujúcimi výslednú kvalitu existuje možnosť vyhodnotia viacrozmerného pozorovania, kde sú jednotlivé podskupiny brané ako ukazovatele a sú zlúčené v analýze Hotellingovho diagramu.

Grafy sú realizované použitím knihovne `matplotlib.pyplot` pre Python a modulom `Canvas` pre JavaScript. Na vyhodnotenie prevedieme výsledné hodnoty (priamo hodnoty pri Shewhartovom diagrame a hodnoty T^2 rozloženia pri Hotellingovom diagrame) do jednorozmerného poľa. Toto pole je následne možné jednoducho vykresliť a pridať farebne odlíšené kontrolné hranice.

5. Analytický výstup štatistickej analýzy
   Medzi posledné vyhodnocované parametre patrí zoznam porušení štatistickej stability na základe pravidiel definovaných v tabuľke 2.1. Pravidlá sú zisťované prehľadávaním a zisťovaním počtov bodov, ktoré stanovené pravidlá porušujú. Na záver je vypísaný zoznam všetkých porušení. V prípade, že nebolo porušené žiadne pravidlo je toto zhodnotenie stability farebne odlíšené.

   Na záver je vyhodnotený index spôsobilosti, ktorý je tiež porovnávaný s príslušnými hodnotami z tabuľky 2.3.

4.2.1 Nadstavba jazyka Python na tvorbu webovej aplikácie

Výsledná aplikácia bola naprogramovaná v jazyku Python, ktorý umožnil zmysluplné prepojenie získavania, spracovania a grafickej reprezentácie dát.

Knižnica *Flask* pracuje najmä s triedou Flask, ktorej inštancia tvorí aplikáciu typu WSGI. WSGI, z anglického Web Server Gateway Interface, je jednoduchá dotazovacia konvencia webových serverov, ktorá slúži k preposielaniu požiadaviek do webových aplikácií alebo programov napísaných v jazyku Python.

Po vytvorení požadovanej inštancie triedy Flask, špecifikujeme názov aplikačného modulu, v prípade výslednej aplikácie sa jedná o jedno modulovú architektúru, k čomu slúži parameter `__name__`, ktorý navyše určuje lokalitu vyhľadávania ďalších súborov na chod aplikácie (súbory návrhov stránok v jazyku HTML a statické súbory).

Ďalej je potrebné si uvedomiť, že samotná aplikácia bude zobraziteľná vo webovom prehliadači a bude bežať na lokálnom serveri tzv. *localhost*, na príslušnej URL adrese. V kóde je možné definovať rôzne akcie, pre rôzne pohyby v rámci hierarchickej štruktúry lokálneho serveru. Je možné špecifikovať relatívne adresy URL, ktoré prislúchajú jednotlivým častiam aplikácie. Tieto aplikácie boli počas návrhu rozdelené s ohľadom na dáta, ktoré od používateľa potrebujú a naopak na dáta, ktoré mu ponúkajú.

Používateľ teda zadá alebo vyberie údaje na jednej stránke, ktorá sa viaže na jednu relatívnu URL a je spracovávaná jednou funkciou v rámci globálneho kódu na ovládanie aplikácie v hierarchii Flasku. Po potvrdení svojej akcie je presmerovaný na ďalšiu stránku, ktorá obsahuje vyhodnotenie jeho vstupu.

Po správnej identifikácii jednotlivých stránok je možné ich implementovať ako funkcie, ktoré vracajú odkazy na príslušné HTML súbory. V HTML súboroch sú uložené kostry zobrazení, ktoré reprezentujú výslednú grafickú podobu stránky. Každá funkcia navyše obsahuje dekorátor *route*, viažúci sa na už pomenovanú inštanciu triedy *Flask*, ktorý definuje

spomínanú relatívnu cestu. Pri zavolaní takto dekorovanej cesty sa vykonajú príkazy príslušnej funkcie.

Pre správnu funkcionalitu je potrebné inštanciu triedy *Flask* na záver zavolať pomocou funkcie `.run()`, pričom spustenie celej aplikácie na lokálnom serveri je možné konvenčným zavolaním prostredia Python z príkazového riadku (`python` alebo `python3` podľa hlavnej nainštalovanej verzie jazyka Python na lokálnom zariadení). Následne je možné otvoriť webový prehliadač na adresu *localhost*, ktorá je špecifikovaná aj vo výpise v príkazovom riadku a potom je možné s aplikáciou pracovať.

Pre používateľov jazyka Python je spustenie veľmi intuitívne, nakoľko sa jedná o jednoduché zavolanie prekladu a spustenia na súbor s koncovkou `.py`.

4.2.2 Komunikácia medzi webovým prostredím a ovládacou logikou

Komunikácia v smere odovzdávania informácii z Pythonu do webového prostredia je pomerne jednoduchá. *Flask* umožňuje pri voľbe toho, s ktorým HTML súborom sa bude na konci vyhodnocovania volanej funkcie pracovať, špecifikovať unikátne parametre v podobe argumentov.

Samotná logika sa dá implementovať aj v HTML pomocou konštrukcie `{\% príkaz \%}`. Vďaka tomu vieme v HTML použiť aj jednoduchú podmienku *if*, zloženú podmienku *if else* alebo cykly *while* a *for*. Všetky tieto konštrukcie je potrebné ukončovať pomocou `{\% end príkaz \%}`. Výhodou je aj možnosť použitia stanovených parametrov, ktoré sme do šablóny HTML poslali ako argumenty z Pythonu. Vieme ich použiť ako premenné, ktoré vieme porovnávať, iterovať cez ne alebo ich hodnoty priamo vypisovať.

Ak sa bližšie zamyslíme nad tým, ako je problematika zobrazenia premenných v HTML implementovaná, zistíme, že sa jedná len o správne stanovenie miesta v HTML štruktúre, ktoré môže obsahovať ľubovoľný text alebo dokonca celé konštrukcie značkovacích príkazov.

Je teda možné:

- zobraziť názov súboru ako nadpis 1. úrovne pomocou `<h1>{{ nazov_suboru }}</h1>`, kde je medzi značky definujúce nadpis a jeho úroveň vložený obsah premennej obsahujúci text nadpisu
- na konkrétnom mieste vložiť celý kód pre tabuľku v jazyku HTML pomocou `{{ kod_tabulky }}`, kde je tento kód vložený ako text ale následne interpretovaný webovým prehliadačom ako skupina príkazov jazyka HTML

Komunikácia opačného typu, teda v smere z webového prostredia do prostredia logického spracovania informácií v Pythone, ktorá je pre správny chod aplikácie tiež veľmi dôležitá, ale podstatne komplikovanejšia. Dáta už nechceme len zobraziť, chceme ich naopak od používateľa získať.

Po spomínanom získaní dát je potrebné brať do úvahy, že dáta budú odosielané sieťovými protokolmi, len tak sa totiž dostanú späť do programového prostredia Python, v ktorom sa vyžaduje ich ďalšie vyhodnotenie. Odosielanie dát po sieti môže predstavovať bezpečnostné riziká (napríklad v prípade, ak by boli zobrazované priamo v políčku adresy URL) ale aj komplikácie práce s pamäťou (nepohodlnosť odosielania dát o veľkých objemoch).

Najprínosnejšie riešenie predstavuje použitie formulárov a ich odosielanie protokolom *POST*. Výhodou je, že v rámci kódu v jazyku HTML vieme určiť identifikátory jednotlivých čiastkových informácií ako parameter `name` pri elementoch typu `<input>`. Odoslanie formulára vieme prepojiť s kliknutím na konkrétny objekt napríklad aj pomocou použitia

JavaScriptu. Po odoslaní vieme identifikovať použitý protokol priamo v Pythone, nakoľko sa viaže na práve otvorenú URL, ktorá je spracovávaná jej príslušnou funkciou. Metóda POST odoslala príslušný HTTP stavový kód úspešnej odpovede, ktorý napomáha k rozlíšeniu stavu prichádzajúcich dát z formulára.

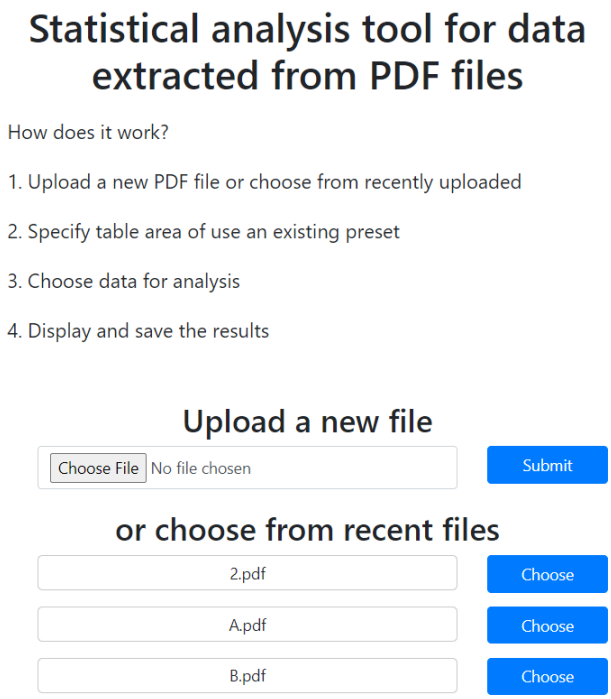
Premenné, ktoré vieme z formulára získať rozlišujeme znakovými reťazcami, ktoré odpovedajú názvom zadaným v parametroch `name` pri elementoch typu `<input>`.

Výmena vstupno-výstupných parametrov je jedným z hlavných kritérií funkčnosti výslednej aplikácie. Vzhľadom na komplikovanejší prístup k výmene určitých typov informácií, je potrebné tieto výmeny realizovať len v nevyhnutných prípadoch a správnym spôsobom.

4.3 Ukážka použitia webovej aplikácie na analýzu dát z PDF súboru

Webová aplikácia pracuje so štyrmi základnými používateľskými krokmi:

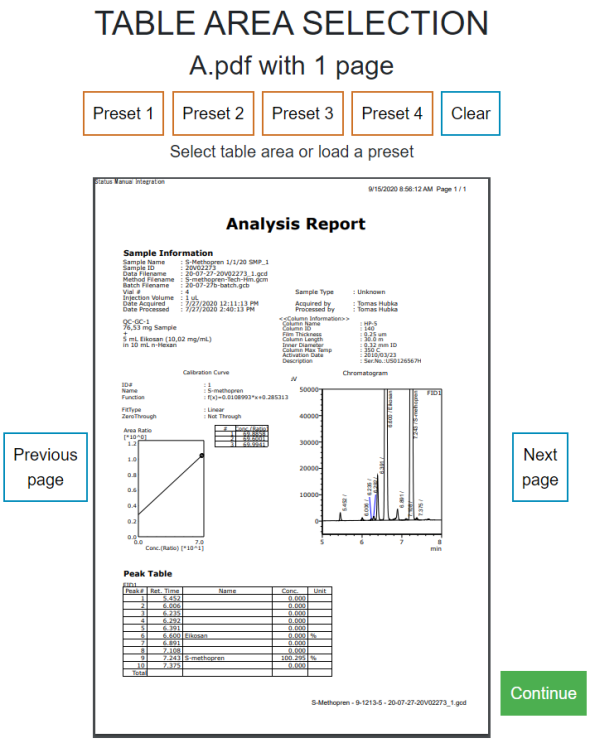
1. Výber a nahratie PDF súboru



Obr. 4.2: Výber PDF súboru nahratím alebo z ponuky.

Používateľovi sú zobrazené základné inštrukcie práce s aplikáciou. Ďalej má ponúknutú možnosť nahratia súboru vo formáte PDF, ktorý má záujem štatisticky spracovať, alebo môže vybrať súbor z histórie posledne spracovaných. Po kliknutí na tlačidlo potvrdenia výberu je presmerovaný na ďalšiu stránku.

2. Výber oblasti tabuľky



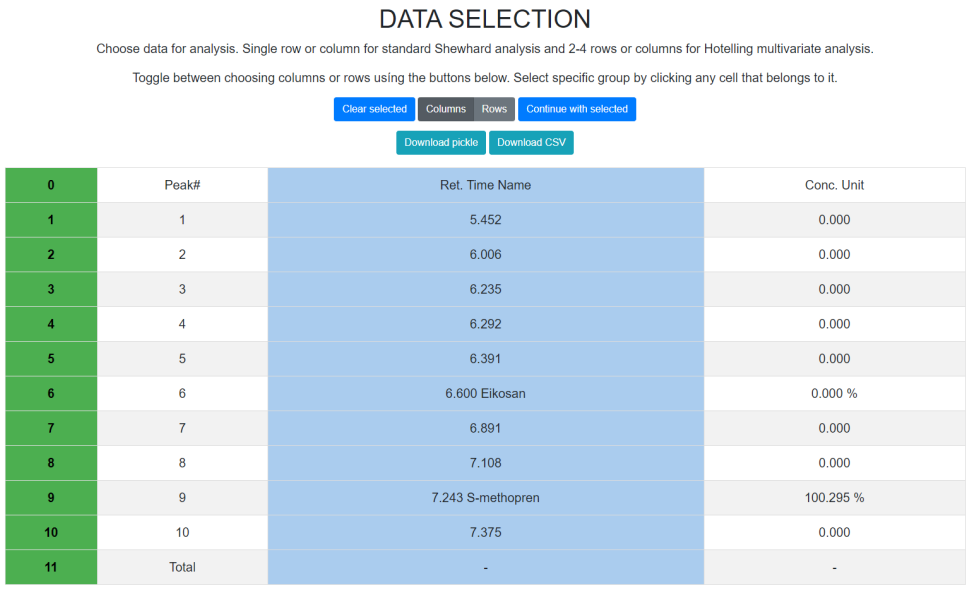
Obr. 4.3: Výber PDF súboru nahratím alebo z ponuky.

Ďalším krokom je špecifikácia oblasti tabuľky v rámci strany PDF dokumentu, tento krok je možné preskočiť, ale je odporúčaný k dosiahnutiu čo najlepších výsledkov. Používateľ má možnosť zvoliť vlastnú oblasť kliknutím a ťahaním myši po oblasti PDF dokumentu, ktorú chce označiť. Ďalšou možnosťou je voľba dopredu prednastavených oblastí, ktoré špecificky vyhovujú potrebám jednotlivých tabuliek v dodanej množine dokumentov. V ukážkovom príklade na obrázku 4.3 vyberáme oblasť ťahaním myši. Možné je vybrať aj oblasť na ľubovoľnej strane. Výber je opäť potvrdený a pokračuje sa na ďalší krok.

3. Výber dát z tabuľky

V tomto kroku je používateľovi umožnený výber dát z tabuľky, ktoré chce analyzovať. Zobrazenú tabuľku vie stiahnuť vo formáte *pickle* a následne v prostredí Python rozbaliť do formátu Pandas DataFrame alebo vo formáte *CSV*, ktorý je ľahko zobraziteľný ľubovoľný softvérom na spracovanie tabuliek. Výber je možný po riadkoch alebo po stĺpcoch, pričom každý výber predstavuje jednu logickú podskupinu. Maximálny možný počet podporovaných podskupín je päť. Výber je opäť potvrdený a používateľ je presmerovaný k prehľadu výsledkov.

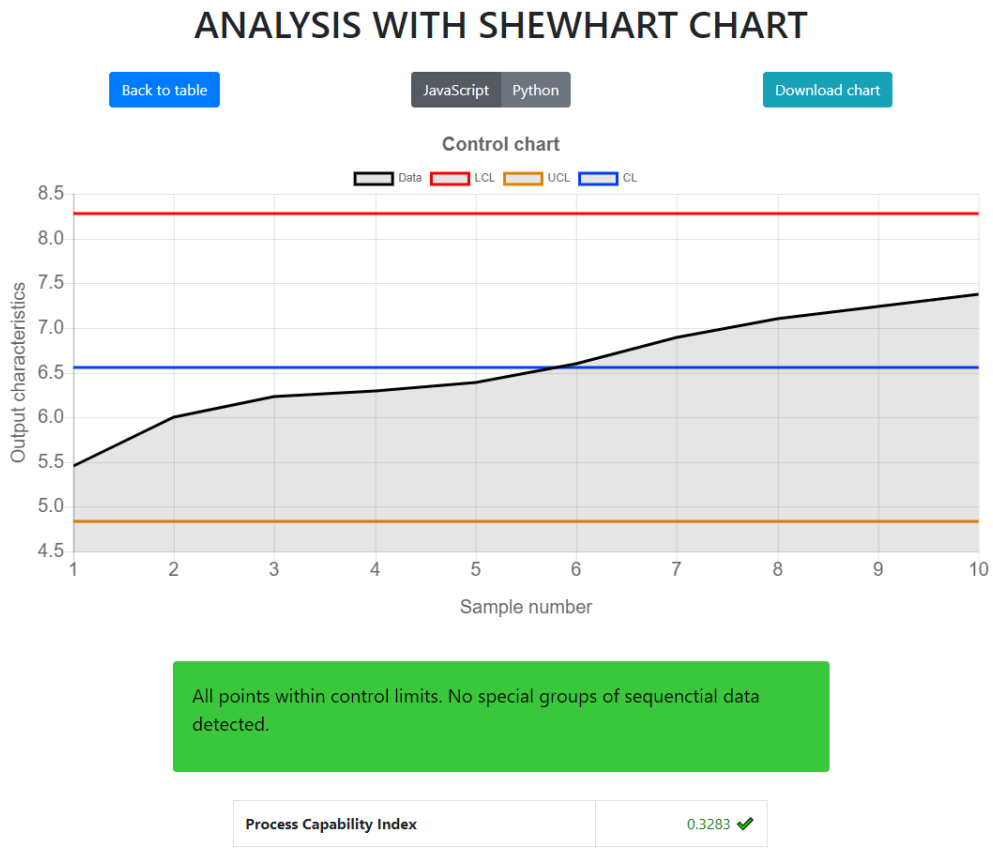
V ukážkovom príklade na obrázku 4.4 vidíme nedokonalosť rozpoznania oproti pôvodnej tabuľke. Je možné sa jej vyhnúť výberom konkrétneho prednastavenia. Výsledok postupu pri výbere predvoleného nastavenia je zobrazený v nasledujúcej kapitole na obrázku 5.6.



0	Peak#	Ret. Time Name	Conc. Unit
1	1	5.452	0.000
2	2	6.006	0.000
3	3	6.235	0.000
4	4	6.292	0.000
5	5	6.391	0.000
6	6	6.600 Eikosan	0.000 %
7	7	6.891	0.000
8	8	7.108	0.000
9	9	7.243 S-methopren	100.295 %
10	10	7.375	0.000
11	Total	-	-

Obr. 4.4: Výber oblasti tabuľky.

4. Zobrazenie štatistickej analýzy



Obr. 4.5: Zobrazenie výsledkov analýzy.

Na záver je používateľovi ponúknutý prehľad vykonanej štatistickej analýzy. Prehľad obsahuje regulačný diagram Shewhartovho alebo Hotellingovho typu, podľa počtu vybraných podskupín v predchádzajúcom kroku. Zobrazenie grafu je možné prepínať medzi grafickým zobrazením pomocou jazykov JavaScript alebo Python. Sú tu možnosti na návrat späť k tabuľke alebo na uloženie grafu ako obrázku vo formáte PNG. Zároveň sa tu nachádza prehľad vyhodnotenia prítomnosti podozrivých zoskupení, ktoré by mohli ohroziť štatistickú stabilitu procesu. V poslednej časti sa nachádza zhodnotenie indexu spôsobilosti. Vyhodnotenia sú farebne odlíšené aby boli rozlíšené prípady stabilného a nestabilného procesu.

4.4 Grafická reprezentácia výsledných grafov

Po úspešnej štatistickej analýze máme k dispozícii všetky dáta, ktoré sme schopní vložiť do súradnicovej sústavy. Potrebné časti grafu, regulačného diagramu, boli už spomínané v kapitole 2 a sú to hlavne pomocné hranice rovnobežné s osou x: UCL a LCL, ale aj naznačenie priemernej strednej hodnoty CL. Tieto osi dotvárajú užívateľsky výhodné zobrazenie zaznamenaných dát a umožňujú rýchle zhodnotenie stability sledovaných procesov.

Výsledné dáta, ktoré máme záujem vykresliť graficky sa nachádzajú v jednorozmernom poli a sú tvorené hodnotami:

- v prípade diagramov Shewhartovho typu: priamo jednotlivé namerané hodnoty
- v prípade diagramov Hotellingovho typu: hodnoty rozloženia T^2 pre definované podskupiny.

Nanesenie týchto dát do súradnicového systému zaznamenávajúceho zmeny hodnôt pre rôzne podskupiny je možné vykonať viacerými spôsobmi.

S prihliadnutím na to, že samotné dáta sú štatisticky analyzované v prostredí *Python*, je samozrejmosťou možnosť ich vyhodnotenia štandardizovanou knižnicou *matplotlib*.

Ak sa ale zamyslíme nad faktom, že výsledné zobrazenie výsledkov sa realizuje vo webovom prostredí, naskytá sa aj alternatíva využitia webu bližších technológií – konkrétne *JavaScriptu*.

Pre dosiahnutie čo najväčšej robustnosti aplikácie a používateľskej slobody sú možné oba prístupy k výslednému grafickému spracovaniu a je teda na používateľovi, ktorý výsledok považuje za vhodnejší a ktorý viac vyhovuje jeho špecifickým potrebám.

Výsledné grafy je navyše možné exportovať aj ako obrázky – to sa už deje výlučne pomocou knižnice *matplotlib*.

Kapitola 5

Testovanie výslednej aplikácie a možné rozšírenia

Testovanie použiteľnosti výslednej aplikácie bolo realizované na množine dodaných PDF súborov. Prvotné testovanie bolo zamerané na samotnú extrakciu dát a spočívalo v nájdení príslušných nastavení pre jednotlivé dokumenty tak, aby boli dosiahnuté čo najlepšie výsledky. Pri testovaní boli overované kombinácie parametrov: *guess*, *area*, *lattice* a *stream*, ktoré sú bližšie popísané v časti 3.2.2. Ideálne parametre boli prevedené do série nastavení viažúcich sa na používateľom voliteľné prednastavenia *Presets*.

Počas testovania bola overovaná správnosť extrakcie ako aj vykonanej štatistickej analýzy.

5.1 Očakávané typy súborov

Nakoľko výsledné riešenie spracováva a vyhodnocuje dáta z dopredu definovanej množiny možných vstupných súborov, je potrebné túto množinu preskúmať a rozdeliť na časti tak, aby dokumenty, ktoré vyžadujú rovnaký spôsob spracovania boli evidované ako príbuzné. Po zistení ideálnych parametrov na extrakciu tabuliek ich môžeme aplikovať na rovnaké vstupné nastavenia.

1. Tabuľka s hlavičkou, bez čiar (viď. tabuľka 5.1)
   Tieto tabuľky nemajú vodiace čiary ohraničujúce bunky tabuľky, čo komplikuje ich automatické rozpoznanie. Spracovanie týchto typov vyžaduje presne vymedzené súradnice výskytu tabuľky (inak splýva s bežným textom). Nakoľko sú výskyty na pozíciách v súboroch odlišné, vyžaduje ich stanovenie manuálnu korektúru (výberom konkrétnej oblasti polohy bunky v dokumente).

2. Ohraničená tabuľka bez stranového presahu (viď. tabuľka 5.2)
   Tabuľky tohto typu sú veľmi vhodné na automatické spracovanie. Okrem čiar exaktnej extrakcii údajov z nich dopomáha aj konzistentnosť ich pozícií v spodnej časti strany, a fakt, že údaje dĺžkou nepresahujú na ďalšiu stranu.

3. Tabuľka v troch stĺpcoch so stranovým presahom (viď. tabuľka 5.3)
   Tabuľka tohto typu je takisto dostatočne strojovo rozpoznateľná, jej tri stĺpce majú pevnú šírku a dodržujú vďaka tomu predpokladaný stanovený tvar, aj napriek absencii čiar. Presah tabuľky na ďalšiu stranu PDF dokumentu bol bez problémov identifikovateľný.

4. Dáta len v jednom stĺpci (viď. tabuľka 5.4)
   Tieto dáta by boli len ťažko rozpoznateľné, nakoľko nie sú v stĺpcoch o pevnej šírka a ani nie sú graficky ohraničené čiarami. K ich úspešnému rozpoznaniu prispieva ale fakt, že vieme presne očakávať, na ktorej pozícii v dokumente sa vyskytujú.

Signal: DAD1A,Sig=258,4 Ref=off

RT [min]	Type	Width [min]	Area	Height	Area%	Name	Peak Theoretical Plates USP	Peak Resolution USP	Peak Tail Factor
2.350	MM m	0.05	1.69	0.53	0.00	BSA	12309.69645		1.41135
4.392	BM m	0.22	37598.36	2555.74	99.99	MBS	2535.17309	9.43303	0.64566
5.528	MM t	0.10	0.43	0.06	0.00		17520.86779	4.38997	0.75119
6.254	MM m	0.14	1.82	0.19	0.00		9546.36331	3.43406	0.90333
		Sum	37602.29						

Obr. 5.1: Ukážka tabuľky s hlavičkou, bez čiar.

Peak Table

FID1

Peak#	Ret. Time	Name	Conc.	Unit
1	5.452		0.000	
2	6.006		0.000	
3	6.235		0.000	
4	6.292		0.000	
5	6.391		0.000	
6	6.600	Eikosan	0.000	%
7	6.891		0.000	
8	7.108		0.000	
9	7.243	S-methopren	100.295	%
10	7.375		0.000	
Total				

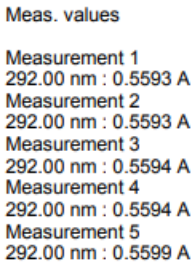
Obr. 5.2: Ukážka ohraničenej tabuľky bez stranového presahu.

Meas. values

Sample 1

200.00 nm : 0.9707 A	200.10 nm : 0.9684 A	200.20 nm : 1.0020 A
200.30 nm : 1.0718 A	200.40 nm : 1.0749 A	200.50 nm : 1.1534 A
200.60 nm : 1.1329 A	200.70 nm : 1.1182 A	200.80 nm : 1.1671 A
200.90 nm : 1.1874 A	201.00 nm : 1.1943 A	201.10 nm : 1.2216 A
201.20 nm : 1.2404 A	201.30 nm : 1.2550 A	201.40 nm : 1.2519 A
201.50 nm : 1.2595 A	201.60 nm : 1.2625 A	201.70 nm : 1.2591 A
201.80 nm : 1.2490 A	201.90 nm : 1.2707 A	202.00 nm : 1.2584 A
202.10 nm : 1.2478 A	202.20 nm : 1.2559 A	202.30 nm : 1.2513 A
202.40 nm : 1.2465 A	202.50 nm : 1.2368 A	202.60 nm : 1.2300 A
202.70 nm : 1.2223 A	202.80 nm : 1.2115 A	202.90 nm : 1.2043 A
203.00 nm : 1.1981 A	203.10 nm : 1.1910 A	203.20 nm : 1.1829 A
203.30 nm : 1.1756 A	203.40 nm : 1.1645 A	203.50 nm : 1.1572 A
203.60 nm : 1.1453 A	203.70 nm : 1.1378 A	203.80 nm : 1.1275 A
203.90 nm : 1.1186 A	204.00 nm : 1.1109 A	204.10 nm : 1.1038 A
204.20 nm : 1.0905 A	204.30 nm : 1.0838 A	204.40 nm : 1.0768 A
204.50 nm : 1.0680 A	204.60 nm : 1.0587 A	204.70 nm : 1.0512 A
204.80 nm : 1.0434 A	204.90 nm : 1.0362 A	205.00 nm : 1.0265 A
205.10 nm : 1.0205 A	205.20 nm : 1.0126 A	205.30 nm : 1.0048 A
205.40 nm : 0.9977 A	205.50 nm : 0.9908 A	205.60 nm : 0.9834 A

Obr. 5.3: Ukážka tabuľky v troch stĺpcoch so stranovým presahom.



Obr. 5.4: Ukážka tabuľky s dátami len v jednom stĺpci.

5.2 Priebeh testovania

Testovanie prebiehalo na vyššie spomínaných typoch tabuliek, ktoré sa nachádzali v dodaných dokumentoch. Súbory boli nahrávané postupom predstaveným v podkapitole 4.3. Testovala sa schopnosť extrakcie správnych údajov z tabuliek a konkrétne pri prednastaveniach boli aplikované dodatočné úpravy tak, aby tabuľkové dáta dávali čo najväčší zmysel.

5.2.1 Overenie správnosti extrakcie dát z PDF súborov

Graficky overujeme výsledky extrakcie a hodnotíme nastavenia pre dané typy tabuliek:

1. Spracovanie tabuľky 1. typu (viď. 1) Tabuľka bola spracovaná správnou technikou tak, aby boli zachované dvojriadkové hlavičky a prázdne miesta.

2. Spracovanie tabuľky 2. typu (viď. 2) Pri tejto tabuľke sa kládol dôraz na správne rozdelenie stĺpcov, nakoľko stĺpce blízko pri sebe mohli splývať.

3. Spracovanie tabuľky 3. typu (viď. 3) Pri spracovaní tretieho typu tabuľky, ktorá je úplne bez čiar, je potrebné rozdeľovať dáta do dvoch stĺpcov, rozdelením tam, kde je znak dvojbodky.

4. Spracovanie tabuľky 4. typu (viď. 4) Pri spracovaní tabuliek posledného typu vynechávame v stĺpci každý druhý riadok, keďže obsahuje len číslo merania. Rovnako použijeme rozdelenie do dvoch stĺpcov ako v predchádzajúcom kroku.

0	RT [min]	Type	Width [min]	Area	Height	Area%	Name	Peak Plates Per Meter USP	Peak Resolution USP	Peak Tail Factor
1	2.350	MM m	0.05	1.69	0.53	0.00	BSA	49238.78578	-	1.41135
2	4.392	BM m	0.22	37598.36	2555.74	99.99	MBS	10140.69236	9.43303	0.64566
3	5.528	MM t	0.10	0.43	0.06	0.00	-	70083.47117	4.38997	0.75119
4	6.254	MM m	0.14	1.82	0.19	0.00	-	38185.45322	3.43406	0.90333
5	-	-	Sum	37602.29	-	-	-	-	-	-

Obr. 5.5: Ukážka výslednej tabuľky získanej po spracovaní oblasti na obrázku 5.1.

0	Peak#	Time	Name	Conc.	Unit
1	1	5.452	-	0.000	-
2	2	6.006	-	0.000	-
3	3	6.235	-	0.000	-
4	4	6.292	-	0.000	-
5	5	6.391	-	0.000	-
6	6	6.600	Eikosan	0.000	%
7	7	6.891	-	0.000	-
8	8	7.108	-	0.000	-
9	9	7.243	S-methopren	100.295	%
10	10	7.375	-	0.000	-
11	Total	-	-	-	-

Obr. 5.6: Ukážka výslednej tabuľky získanej po spracovaní oblasti na obrázku 5.2.

1	200.00 nm	0.9707 A	200.10 nm	0.9684 A	200.20 nm	1.0020 A
2	200.30 nm	1.0718 A	200.40 nm	1.0749 A	200.50 nm	1.1534 A
3	200.60 nm	1.1329 A	200.70 nm	1.1182 A	200.80 nm	1.1671 A
4	200.90 nm	1.1874 A	201.00 nm	1.1943 A	201.10 nm	1.2216 A
5	201.20 nm	1.2404 A	201.30 nm	1.2550 A	201.40 nm	1.2519 A
6	201.50 nm	1.2595 A	201.60 nm	1.2625 A	201.70 nm	1.2591 A
7	201.80 nm	1.2490 A	201.90 nm	1.2707 A	202.00 nm	1.2584 A
8	202.10 nm	1.2478 A	202.20 nm	1.2559 A	202.30 nm	1.2513 A
9	202.40 nm	1.2465 A	202.50 nm	1.2368 A	202.60 nm	1.2300 A
10	202.70 nm	1.2223 A	202.80 nm	1.2115 A	202.90 nm	1.2043 A
11	203.00 nm	1.1981 A	203.10 nm	1.1910 A	203.20 nm	1.1829 A
12	203.30 nm	1.1756 A	203.40 nm	1.1645 A	203.50 nm	1.1572 A

Obr. 5.7: Ukážka výslednej tabuľky získanej po spracovaní oblasti na obrázku 5.3.

Measurement 1	292.00 nm	0.5593 A
Measurement 2	292.00 nm	0.5593 A
Measurement 3	292.00 nm	0.5594 A
Measurement 4	292.00 nm	0.5594 A
Measurement 5	292.00 nm	0.5599 A

Obr. 5.8: Ukážka výslednej tabuľky získanej po spracovaní oblasti na obrázku 5.4.

5.2.2 Ukážky zaujímavých postupov využitých pri testovaných prípadoch

V priebehu testovania bola overovaná schopnosť správneho fungovania aplikácie, pre špecifické komplikovanejšie prípady:

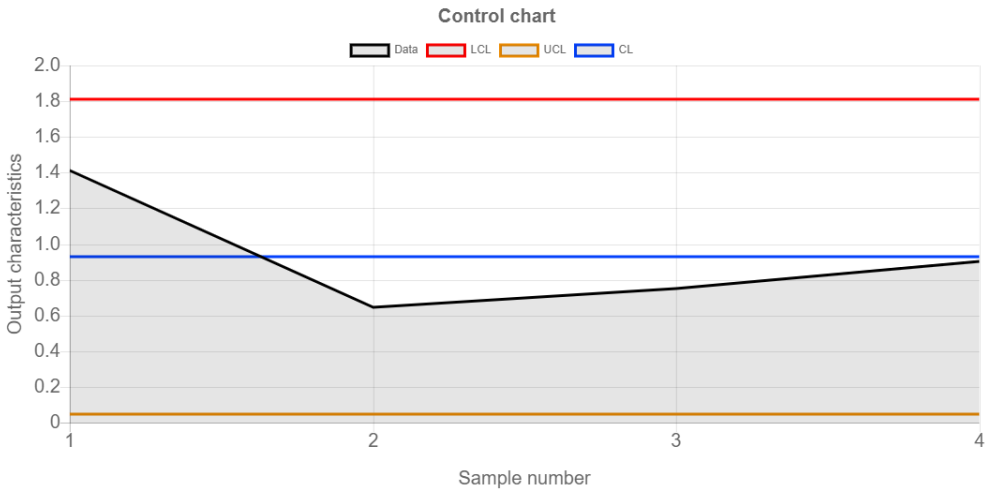
1. Odstránenie prázdnych hodnôt
   Výskyt prázdnych hodnôt môžeme pozorovať napríklad pri tabuľke na obrázku 5.1. Vynechané prázdne miesta je potrebné špecificky označiť a ukladať v DataFrame aby neovplyvnili výsledky analýzy. Ukážka výslednej tabuľky získanej z pôvodného dokumentu použitím *Preset 4* a výberom konkrétnej oblasti z dokumentu (obrázok 5.1).

2. Pretypovanie na desatinné čísla
   Po získaní dát z tabuliek je obsah každej bunky vo formáte `string`. Na matematickú analýzu potrebujeme tieto dáta mať vo formáte `float`, čo dosiahneme pretypovaním celého stĺpca a vylúčením nechcených hodnôt.

3. Odstránenie nechcených hodnôt – písmen pri inak číselných dátach
   Oddelenie čísla od písmen (napríklad názvu jednotky) realizujeme s využitím regulárnych výrazov, ktoré ignorujú nečíselné hodnoty.

5.2.3 Testovanie výsledkov štatistickej analýzy

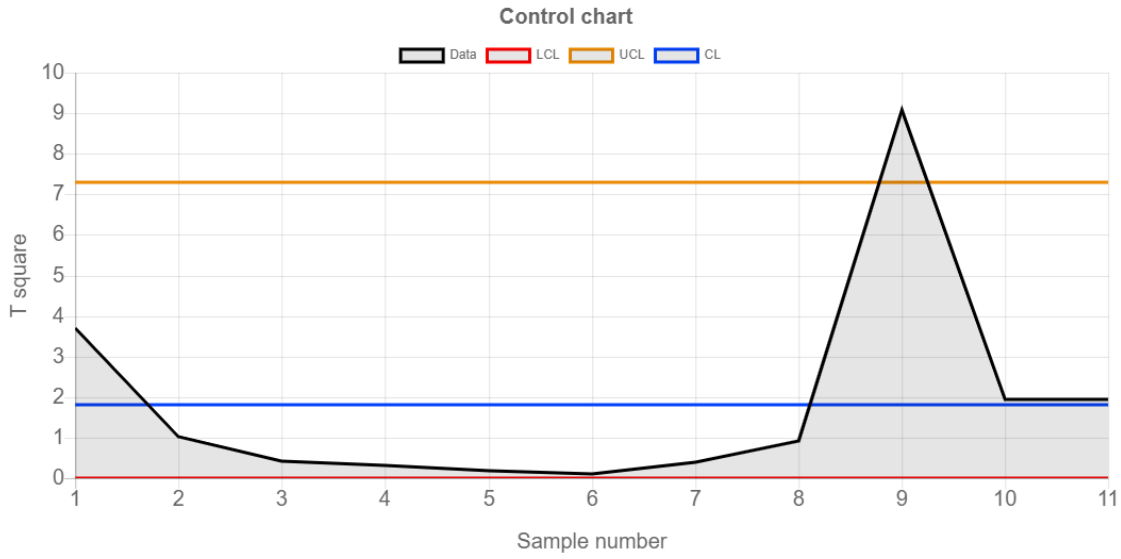
Dáta získané z tabuliek vieme následne spracovať. Výsledok spracovania závisí od množstva vybraných dát. Pre jednu sledovanú podskupinu realizujeme jednorozmerné štatistické zhodnotenie pomocou Shewhartovho diagramu. Pri viacerých znakoch, sa používa Hotellingov diagram. Pri testovaní analyzovania rôznych typov tabuliek sme získali nasledovné výsledky:



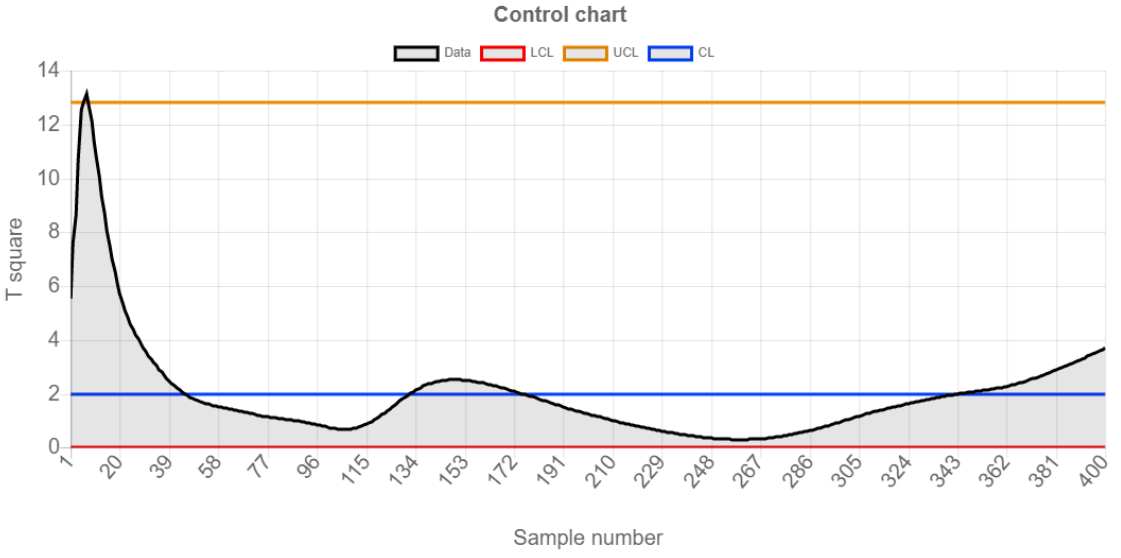
All points within control limits. No special groups of sequenctial data detected.

Process Capability Index	0.0863 ✔
Lower Limit	0.0863 ✔
Upper Limit	0.0863 ✔

Obr. 5.9: Analýza so Shewhartovým diagr., vykonaná z posledného stĺpca z obr. 5.1 a 5.5.



Obr. 5.10: Analýza s Hotellingovým diagr., vykonaná z 2. a 4. stĺpca z obr. 5.2 a 5.6.



Obr. 5.11: Ukážka Hotellingovho diagr., zostrojeného z 3. a 4. stĺpca z obrázkov 5.3 a 5.7.

Special causes detected:

Point number 7 outside of control limits.

Action limit: 2 out of 3 consecutive points in action zone.

Inner limit: at least 7 consecutive points detected on one side of CL.

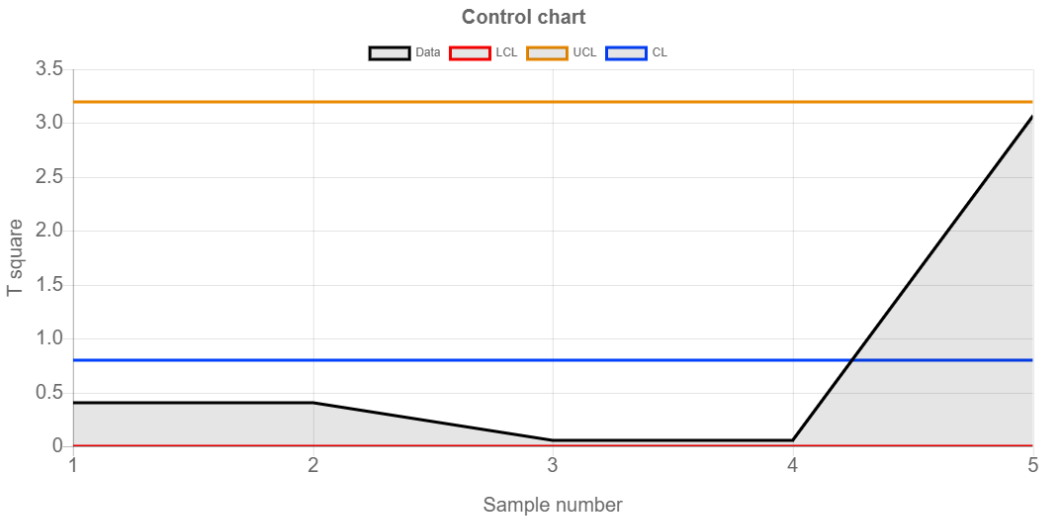
Warning limit: 4 out of 5 consecutive points in warn zone.

Too much variability: 8 consecutive points outside of inner zone.

Not enough variability: 15 consecutive points inside inner zone.

Process Capability Index	1.0695 ✔
Lower Limit	1.3292 ✘
Upper Limit	7.2165 ✘

Obr. 5.12: Ukážka analýzy vykonanej z 3. a 4. stĺpca z obrázkov 5.3 a 5.7.



All points within control limits. No special groups of sequenctial data detected.

Process Capability Index	0.4643 ✔
Lower Limit	0.3058 ✔
Upper Limit	0.9157 ✔

Obr. 5.13: Analýza s Hotellingovým diagr., vykonaná z oboch stĺpcov z obr. 5.4 a 5.8.

5.3 Možné budúce rozšírenia

Aplikácia môže byť v budúcnosti rozšírená. Ako najzmysluplnejšie sa ponúka rozšírenie v oblasti prenesenia aplikácie na server, aby bola k dispozícii online. Toto možné rozšírenie vyžaduje implementáciu databáz na ukladanie PDF, súborov, tabuľkových dát a grafov. Na serveri by bolo následne možné nastaviť automatické spúšťanie aplikácie *Flask*. Netreba zabudnúť na doinštalovanie všetkých potrebných nástrojov a knižníc.

Ďalšie možné rozšírenie spočíva v ponúknutí väčšieho množstva analýz, z ktorých by si používateľ mohol vyberať a taktiež vo vylepšení systému rozpoznávania tabuliek na neznámych oblastiach v dokumente.

Takisto je možné rozšíriť aplikáciu o možnosti editácie tabuľky prevedenej z PDF dokumenetu. Možnosti ako: rozdelenie jedného stĺpca na dva, na základe vyskytujúceho sa znaku, prepis alebo dopísanie neznámych hodnôt, alebo vynechanie konkrétnych riadkov by mohlo zvýšiť celkovú robustnosť aplikácie.

Celkovo je riešenie možné rozširovať v oblastiach získavania dát, ich exportu a ďalšieho spracovania. Riešenie ponúka zmysluplné možnosti nadstavieb, nakoľko sa jeho postup dá rozdeliť na konkrétne kroky s medzivýsledkami, pričom tieto medzivýsledky môžu byť v rámci prípadných rozšírení vyhodnocované iným spôsobom.

Záver

V tejto bakalárskej práci som štatisticky analyzovala dáta získané z PDF súborov. Podarilo sa mi nadobudnúť dostatočné teoretické znalosti z oblasti ukladania, spracovania a extrakcie dát z PDF súborov, ako aj matematické základy z oblasti štatistickej regulácie procesov. Tieto znalosti sa mi podarilo aplikovať do výslednej webovej aplikácie, ktorá využíva techniky extrakcie dát a následne ich štatisticky analyzuje.

Podarilo sa mi splniť cieľ úspešného spracovania dát z dodaných PDF súborov. Výsledná aplikácia je schopná vyhľadávať tabuľky, a to buď ich automatickou detekciou, alebo vyznačením konkrétnych, užívateľom označených oblastí. Následne je z nich schopná exportovať tabuľkové dáta a uložiť ich do vhodnej podoby v prostredí Python.

Aplikácia prináša najlepšie výsledky s využitím prednastavení aplikovaných na dokumenty typov, s ktorými sa počítalo pri návrhu a implementácii tejto aplikácie. Poskytuje používateľovi prehľadné kroky a výsledné výstupy v podobe grafov a potrebných kontrolných hodnôt.

Literatúra

[1] Adobe. *What is PDF?* [online]. Adobe, 2021 [cit. 2020-04-12]. Dostupné z: https://acrobat.adobe.com/sk/sk/acrobat/about-adobe-pdf.html.

[2] Aristarán, M. a Tigas, M. *Introducing Tabula* [online]. OpenNews, 2013 [cit. 2020-04-22]. Dostupné z: https://source.opennews.org/articles/introducing-tabula/.

[3] Dunn, K. *Shewhart charts* [online]. learncache.org, 2010 [cit. 2020-04-09]. Dostupné z: https://learnche.org/pid/process-monitoring/shewhart-charts#shewhart-charts.

[4] engineering crossML. *Extracting data from PDF documents* [online]. Medium, 2020 [cit. 2020-04-17]. Dostupné z: https://medium.com/crossml/extracting-data-from-pdf-documents-7792d5c1e403.

[5] Floreková, L. The statistical process control methods - SPC. *Acta Montanistica Slovaca*. Marec 1998, zv. 3, s. 21.

[6] Heckert, A. *HOTELLING CONTROL CHART* [online]. NIST, 2015 [cit. 2020-04-13]. Dostupné z: https://www.itl.nist.gov/div898/software/dataplot/refman1/auxillar/hotell.htm.

[7] Jarošová, E. a Noskievičová, D. *Pokročilejší metody statistické regulace procesu*. 1. vyd. Grada, 2015. ISBN 978-80-247-5355-3.

[8] Krpenská, M. *Využití regulačních diagramů v monitoringu kvality zdravotního screeningu [online]*. 2014 [cit. 2021-04-13]. Dostupné z: https://is.muni.cz/th/qn0vh/.

[9] Kumar, S. *Extract Tables from PDF file in a single line of Python Code* [online]. Towards Data Science, 2021 [cit. 2020-04-24]. Dostupné z: https://towardsdatascience.com/extract-tables-from-pdf-file-in-a-single-line-of-python-code-5b572cd9fbe5.

[10] McNeese, B. *Control Chart Rules and Interpretation* [online]. BPI Consulting, 2016 [cit. 2020-04-12]. Dostupné z: https://www.spcforexcel.com/knowledge/control-chart-basics/control-chart-rules-interpretation.

[11] Meloun, M. et al. *Kompendium statistického zpracování dat*. 1. vyd. Academia, 2002. ISBN 8020010084.

[12] Montgomery, D. C. *Introduction to Statistical Quality Control*. 6. vyd. John Wiley and Sons, Inc, 2008. ISBN 978-0-470-16992-6.

[13] NIST/SEMATECH. *What is Process Capability?* [online]. NIST/SEMATECH e-Handbook of Statistical Methods, 2012 [cit. 2020-04-15]. Dostupné z: https://www.itl.nist.gov/div898/handbook/pmc/section1/pmc16.htm.

[14] Noskievičová, D. a Tošenovský, J. *Statistické metody pro zlepšování jakosti.* 1. vyd. Montanex, 2000. ISBN 807225040X.

[15] Parker, T. *PDF Page Coordinates* [online]. WindJack Solutions, 2008 [cit. 2020-04-22]. Dostupné z: https://www.pdfscripting.com/public/PDF-Page-Coordinates.cfm.

[16] Richterová, S. *Využití statistických metod ve výrobním procesu společnosti Tegü VUKO, spol. s r.o. [online].* 2010. Dostupné z: http://hdl.handle.net/10563/14105.